NATIONAL RADIO ASTRONOMY OBSERVATORY
GREEN BANK, WEST VIRGINIA

ELECTRONICS DIVISION INTERNAL REPORT No. 221

JPL PHYSICAL OPTICS SCATTERING PROGRAM

JAMES R. LYONS

SEPTEMBER 1981

NUMBER OF COPIES: 150

JPL PHYSICAL OPTICS SCATTERING PROGRAM

James R. Lyons

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED):

JPL PHYSICAL OPTICS SCATTERING PROGRAM

James R. Lyons

I. INTRODUCTION

This report describes a FORTRAN program developed at JPL for the calculation of the scattered pattern from a reflector of arbitrary shape. The program written by A. Ludwig is contained in a report entitled "Calculation of Scattered Patterns from Asymmetrical Reflectors" (Technical Report 32-1430; 2/15/70). The JPL report stresses the following: the use of a Fourier expansion representation of the reflector surface, the possible inaccuracy of a far field assumption for the fields incident on the reflector, and the use of a fast integration technique developed by Ludwig. Some good examples of the accuracy of the program are included. A copy of the JPL report may be obtained from Rick Fisher in Green Bank or from Sarah Martin (microfiche) in Charlottesville.

The program is actually two programs: one which calculates a scattered pattern and one which computes, using a spherical wave expansion, a completely general representation of the incident field pattern. The programs will be referred to as SCAT and SWE, respectively.

The method of surface specification has been altered to that of tabular form and the integration routine has been slightly adjusted to better handle reflector edge contributions. No major changes have been made to the SWE program.

SCAT employs the technique of physical optics to calculate the scattered field. The program takes the Fourier transform, using a fast integration technique, of an array of induced current dipoles.

The incident field representation provided by the SWE program allows accurate calculations to be made even with the reflector well into the near field of the source. Input into the SWE program is the far field source pattern, such as those typically measured on an antenna range. The far field form of spherical waves is matched to the (far field) incident pattern. Backing up into the near field, the SWE accurately approximates the transverse and radial components of the electric field.

(An attempt has been made towards the end of this report to describe in physical terms some of the theory behind the SCAT and SWE programs.)

One application of this program is in checking the taper at the edge of the main dish of the 140-ft Cassegrain telescope. For this problem the scattering surface is the subreflector, which has an asymmetrical edge, and the incident field is the antenna range (or computed) pattern of the appropriate Cassegrain feed.

A slight variation of this situation involves the re-design of the subreflector to include a conical vertex plate and a flange about the reflector edge.

Another application is to aid in the design of beam polarization splitters which may be added to the 140-ft. To the SCAT program, the beam splitter would be a tilted plane reflector placed very near the feed aperture.

II. GENERAL PROGRAM CHARACTERISTICS

A. Summary of What Program Does

1. SCAT program.

Given a scattering surface specified by $\rho(\theta,\phi)$ (i.e., spherical coordinates) and the magnetic fields incident on this surface, the scattered pattern is computed over a grid of observation points. The scattering surface is assumed to be perfectly conducting; therefore, the electric field is zero at the surface.

The values of ρ are either input in tabular form or calculated from an equation in θ and/or ϕ.

The E and H plane values (dB or volts, and degrees) of the incident magnetic field are read in by the SWE program. The SWE program computes the coefficients of the expansion and stores them. These stored values are read in by the SCAT programs.

Ludwig has shown that assuming far field conditions for the source field for the trivial case of scattering from an infinite plane reflector located within or near the traditional far field boundary line ($2D^2/\lambda$, D = source diameter) can result in rather strong backlobes in the scattered pattern. The strength of the backlobe and distortion of the main (reflected) lobe increases as the reflector further penetrates the near field; in other words, the magnitude of the error depends on frequency. (See JPL report, figures 3-5 on pages 6-8.) Ludwig further shows (Figure 7) that use of the spherical wave representation almost entirely eliminates the problem.

The scattered fields are, as mentioned above, computed using the method of physical optics. It is assumed that the induced currents are zero on shadowed portions of the reflector and that on directly illuminated areas the induced surface current value is twice the value of the tangential (to the surface) incident H field. These two assumptions constitute the physical optics approximations.

As a means of reducing computation time the far field form of the scattered pattern is calculated; the near field form can be found by submitting the scattered field values to the SWE program.

Summing the product of the induced surface currents and phase delay (pathlength) for each Δs of the surface, the scattered field as seen from a particular point on the output grid is determined. This integration is performed for each point on the output grid. Finally, the far field source pattern is added to the scattered pattern yielding what is termed the "total fields". For the situation of an infinite plane reflector, the total fields would be the incident fields pointed in the opposite direction on the output grid and with a 180° phase change.

From comparisons of computed and measured subreflector patterns, Ludwig has shown the computed pattern to be accurate down to -35 dB (relative power) and even through the first side lobe. Fairly reliable spillover efficiency calculations should be possible, since the major spillover contribution is generally from fields within 25 dB of the pattern maximum. Proper handling of the side lobes requires the use of the Geometrical Theory of Diffraction (GTD) which does not make the assumption of zero induced currents on shadowed reflector surfaces.

2. SWE program.

The SWE program finds the coefficients of expansion of the incident (far zone) magnetic field pattern in terms of transverse electric (TE) and transverse magnetic (TM) spherical waves. The TE and TM spherical waves are the general vector solutions to Maxwell's equations for an electromagnetic wave travelling in a source-free region, V. For the case of representing a magnetic field, the TM vector solution has no radial components, whereas the TE solution has components in all three coordinate directions.

A spherical wave expansion operates under the same mathematical principles as the Fourier expansion of a function. In a Fourier expansion, a function is represented by a summation of sines and cosines each multiplied by a coefficient particular to the order of sine and cosine variation; this order of variation could be called the mode order. To represent the function, the two sets of coefficients (one set multiplying the sines, the other the cosines) must be determined. This is done by evaluating integrals involving sines and cosines and specific function values. The integrals are simplified through the use of the orthogonality relations between sines and cosines.

In a spherical wave expansion the sines and cosines of the Fourier expansion are replaced with the TE and TM spherical wave solutions of Maxwell's equations. The integrals for the mode coefficients contain the corresponding TE and TM mode functions and the far field incident pattern values. Thus, the procedure for determining the coefficients of the spherical wave expansion is basically the same as for that of the Fourier expansion.

The expansion is in three variables, in this case the spherical coordinates, and contains two mode orders. Thus, the expansion is a double summation; one summation for each mode variable. One mode order, generally called the mode order and designated by a "n", specifies the degree of ρ variation. The other mode order, called the order of azimuthal variation and designated by an "m", specifies the degree of ϕ variation. The degree of θ variation depends on both mode orders.

In principle, the expansion scheme outlined above can represent any input pattern at any distance from its source. However, because the case of $m = 1$ is of particular importance, the double summation has been reduced in the program to a single summation over n. Also, one set of spherical wave solutions has been neglected, which means that the incident radiation is assumed to be linearly polarized.

B. Coordinate System Used.

The origin of the system will be at the phase center of the source fields. The reflecting surface is specified by the vector $\bar{\rho}$ which is a function of the angles θ and ϕ; ρ, θ, ϕ represent a point on the surface in spherical coordinates. Since $\bar{\rho}$ is defined by θ and ϕ, these two angles will specify the point of integration.

The output grid over which the observer views the scattered pattern is also designated by spherical coordinates (R, Θ, Φ). Since the far field form of the scattered pattern is computed, the output grid is defined entirely by Θ and Φ.

The Z-axis is the reflector axis. The three coordinate sets — (ρ, θ, ϕ), (R, Θ, Φ) and (X, Y, Z) — all have the same origin. The coordinate system is shown in Figure 1.

C. Program Structure and Order of Calculation.

1. SCAT program.

For storage reasons, the reflector surface over which the integration is performed is divided into integration grids. The scattered fields from each grid are superimposed and added to the incident fields to yield the total (scattered) pattern.

The reflector is divided along θ and/or ϕ (Figure 2), depending on the size of the reflector, the choice of $\Delta\theta$ and $\Delta\phi$ and storage requirements. This partitioning introduces no appreciable error and is quite useful since the total integration may cover several thousand points.

The output grid is not segmented because it generally includes only several hundred points at most.

For each integration grid, $\rho(\theta, \phi)$ is either read in or calculated for every (θ, ϕ) point on the grid. Since the normal to the surface is required at each integration point, $\frac{\partial\rho}{\partial\theta}(\theta, \phi)$ and $\frac{\partial\rho}{\partial\phi}(\theta, \phi)$ are computed over the grid. The derivatives are computed using the numerical method of backward and forward differences or by an analytic expression in θ and ϕ which is inserted in subroutine SURF.

In general the outermost (in θ) integration grids will contain points that are beyond the edge of the reflector. This is so because in the program, θ and ϕ are independent of each other; that is, the integration grid is represented by two vectors rather than by a matrix of points. However, at the reflector edge, the boundary is defined by θ_{edge}, which will generally be a function of ϕ. To correctly represent the reflector, the edge values of θ are read in or calculated for each ϕ of the integration grid.

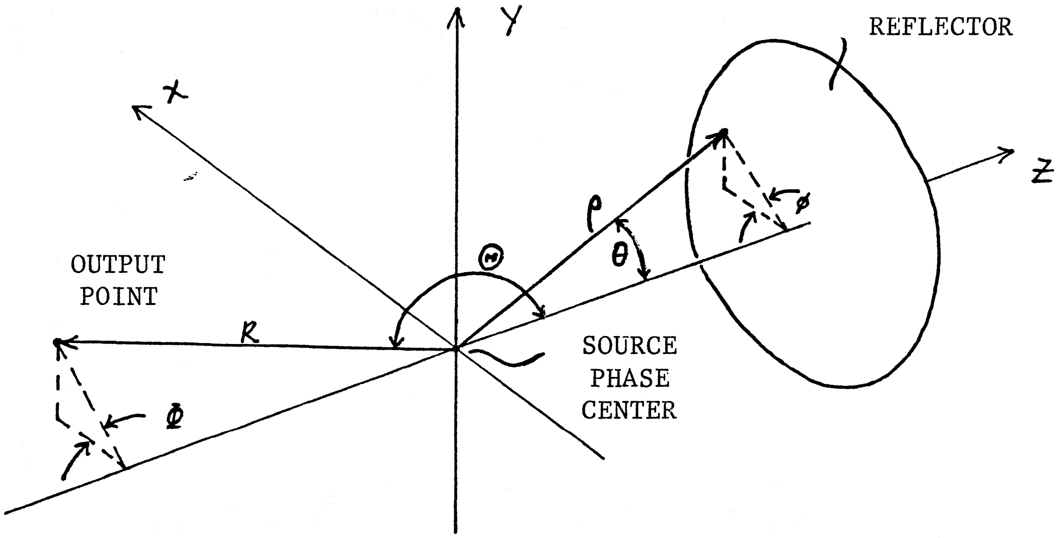


FIGURE 1

Coordinate system (from JPL report, page 3).

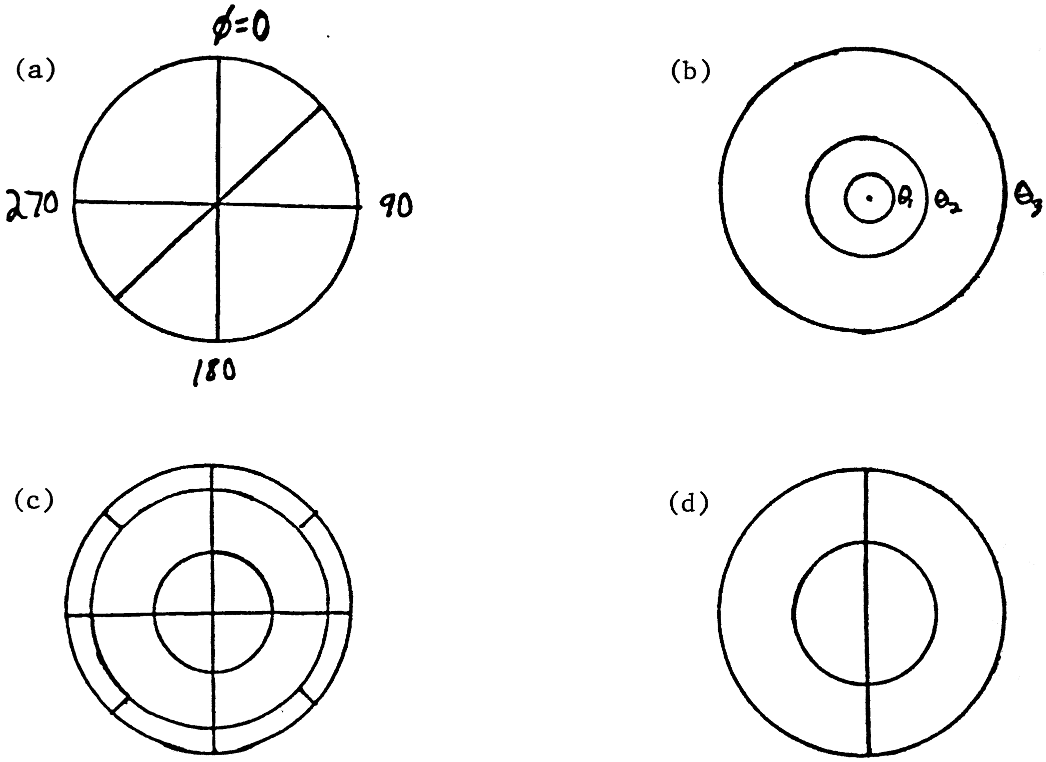


FIGURE 2

Sample integration grids.

An integer parameter, IEDGE, is used to determine if the edge will be encountered for the present integration grid and, if it is encountered, how the edge values will be obtained. IEDGE is read in by the MAIN subroutine of the SCAT program for each integration grid.

As mentioned above, the edge integration grids will contain points beyond the actual reflector. Since such values may not be known, they are not required inputs. When the values of $\rho(\theta, \phi)$ are in tabular form, the user can set to 999.99999 the first ρ value that corresponds to a point beyond the reflector edge; the ρ values remaining for that given value of ϕ can (and must) be set to some arbitrary value (e.g., zero).

If all the values of ρ for an edge integration grid are known, as is the case when ρ is calculated analytically, then including them will result in slightly better accuracy.

Note that the SCAT program is at its maximum user-convenience when ρ (in particular) and the edge values of θ can be computed from equations inserted in the program.

An informal flowchart of the SCAT program is shown in Figure 3. Subroutine names are written next to most flowchart components.

2. SWE program.

The organization of the SWE program is considerably simpler; therefore, no flowchart of it is shown. A mode order term and the far field E and H plane pattern values from the appropriate feed comprise the inputs. Details of input considerations will be given below.

After reading in the input pattern, the program establishes a matrix of θ mode weights (forms of Lengendre polynomials) and a matrix of input pattern differentials. The evaluation of the coefficient integrals is then done by matrix multiplication. The real and imaginary values of the TE and TM coefficients are normalized and stored on the computer disk. As a check on the validity of the coefficients, the far field form is computed allowing a comparison to be made with the input pattern.

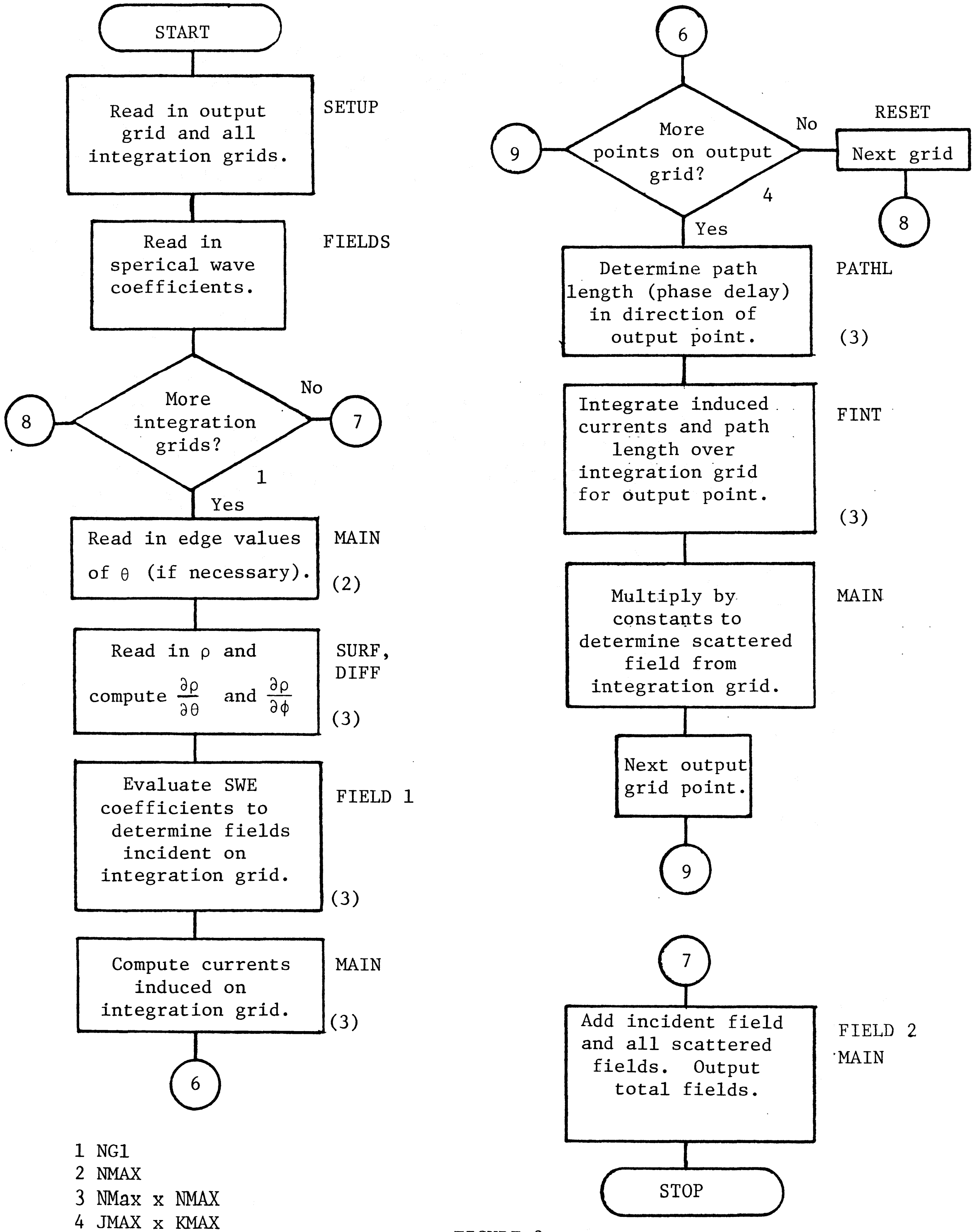


FIGURE 3

SCAT program flowchart.

III. DETAILS OF USING THE PROGRAM

Both programs are stored on the disk of the IBM 360 computer located in Charlottesville. The user command system is called Pandora. Each subroutine is stored as a Pandora member and has a member name either identical to or similar to the subroutine name. The Pandora member name (PMN) will be given for each subroutine.

For both the SCAT program and the SWE program this section provides the following information:

For each subroutine of the program:

Subroutine description, input variables (that READ data), input considerations, and a table of inputs and their formats.

Discussion of program output.

A. Subroutines of Scattering Program

1. MAIN (PMN MAIN1SC)

Subroutine description:

This routine coordinates the efforts of the program. The procedure followed by MAIN has been outlined by the SCAT program flowchart (Figure 3). Note the SUBROUTINE names written by the flowchart blocks.

Input variables and their tasks:

TITLE - any alphanumeric statement $\leq$ 72 characters.

PC - phase constant, $k = 2\pi/\lambda$ (λ in meters).

XT, YT, ZT - translations to expected phase center of scattered pattern (affects only output phase data).

SCALE - scale factor for output fields. If input is zero, program sets SCALE = 1.0.

(The following variables are read in for <u>each</u> integration grid.)

IEDGE - an integer; determines if and how program obtains θ at the reflector edge. There are four cases:

IEDGE (continued):

IEDGE < 0 - Program calculates values for that integration grid using equations inserted into subroutine EDGEEQ.

IEDGE = 0 - θ at edge is constant. Program reads in just one value.

IEDGE = 11 - Edge will <u>not</u> be encountered on present integration grid. No values read in.

IEDGE > 0 ($\neq$ 11) - θ at edge is a function of ϕ and is read in for each value of ϕ in the integration grid (NMAX values).

TEDGE (NMAX) - edge values of θ in degrees. Either NMAX values, one value or no values are read in depending on IEDGE.

Input considerations:

TITLE usually describes reflector and frequency. The scattered field phase center translations may be set to zero if the output phase is of no interest.

The choice of $\Delta\theta$ and $\Delta\phi$ (which determines the number of integration grids) is covered in SETUP.

Table of inputs and their formats:

	Card	Input Variable(s)	Format
	1	TITLE	18A4
	2	PC, XT, YT, ZT, SCALE	5F10.4
		[SETUP DATA]	
		[FIELDS DATA]	
Read in for each integration grid.		IEDGE	I5
		TEDGE (Either NMAX, 1 or no values.)	7(1X, F9.5)
		[SURF DATA]	

2. Setup (PMN SETUP)

Subroutine description:

This routine sets up the output grid and all integration grids. The integration grid parameters are reset for each new grid by ENTRY RESET within SETUP.

The input parameters of the output grid consist of the number of values (JMAX and KMAX), initial value and increment which define the Θ and Φ points over which the fields are viewed. A typical grid could be $\Phi = 0°$ and $90°$ and $\Theta = -90°$ to $+90°$ with a step of $2°$. Thus, KMAX = 2, JMAX = 91.

The input parameters of each input grid are identical to those of the output grid but define θ and ϕ instead. As an example, consider Figure 2(d). The first grid might be $\theta = 0°$ to $6°$ with $\Delta\theta = 0.2°$ and $\phi = 0°$ to $180°$ with $\Delta\phi = 4°$. Thus, MMAX = 31 and NMAX = 46. The next grid might be $\theta = 0°$ to $6°$ and $\phi = 180°$ to $360°$. Two more grids would follow. The order of the four grids is of no consequence.

Input variables and their tasks:

JMAX	-	number of Θ values on output grid, ≤ 181.
TT1	-	initial Θ value.
DDT	-	Θ increment.
KMAX	-	number of Φ values on output grid, ≤ 5.
PP1	-	initial Φ value.
DPP	-	Φ increment.
NG1	-	number of integration grids, ≤ 21.
MM(I)	-	number of θ values on Ith integration grid, ≤ 36.
TI	-	initial θ value.
DT	-	θ increment ($\Delta\theta$).
NN(I)	-	number of ϕ values on Ith integration grid, ≤ 91.
PI	-	initial ϕ value.
DP	-	ϕ increment ($\Delta\phi$).

(All initial values and increments in degrees.)

Input considerations:

$\Delta\theta$ and $\Delta\phi$ are chosen such that the dimensions of the largest ΔS on the reflector surface are roughly one wavelength by one wavelength. This ensures that the pathlength term does not vary by more than 2π over any ΔS. Thus, $\Delta\theta$ can be found from $\Delta\theta \approx \lambda/\rho_{max}$, where ρ_{max} is the greatest distance to the reflector and $\Delta\theta$ is in radians. The corresponding equation for $\Delta\phi$ is $\Delta\phi \approx \Delta\theta/\sin(\theta_{max})$, where θ_{max} is the largest θ angle subtended by the reflector edge.

The values obtained for $\Delta\theta$ and $\Delta\phi$ should be rounded to the nearest convenient value (tenths or hundredths of a degree). An upper limit of $\Delta\theta = 1°$ is necessary, since larger values sample the incident field too sparsely. Decreasing $\Delta\theta$ and/or $\Delta\phi$ offers a good check on the validity of the chosen increments. However, halving either increment nearly doubles the computation time. To give some idea of the CPU time needed for a given JMAX, KMAX, TMMAX, TNMAX combination (define TMMAX and TNMAX as total number of θ and ϕ points on reflector surface), the following table is provided.

KMAX	JMAX	TMMAX	TNMAX	CPU TIME (minutes)	
1	91	35	92	18	
1	91	35	184	36	IBM
1	181	35	184	50	360/65
2	181	35	184	89	

To ensure accurate edge contributions, the user should be certain that for any value of ϕ the edge value of θ does not fall within the θ range of the first two values of θ (inclusive) on the integration grid. For most cases, the edge value of θ could be within this range and would produce no noticeable differences in the total scattered pattern. It is, however, absolutely necessary that all edge θ values be greater than or equal to the first θ of integration.

The integration routine and SURF subroutine set the limit of MMAX and NMAX ≥ 4.

Table of inputs and their formats:

Card	Input Variable(s)	Format
1	JMAX, TT1, DTT	I5, 2F10.2
2	KMAX, PP1, DPP	I5, 2F10.2
3	NG1	I5
4	MM(1), T1, DT	I5, 2F10.2
5	NN(1), P1, DP	I5, 2F10.2
⋮	⋮	⋮
2NG1 + 2	MM(NG1), T1, DT	I5, 2F10.2
2NG1 + 3	NN (NG1), P1, DP	I5, 2F10.2

3. FIELDS (PMN FIELDS)

Subroutine FIELDS reads, off the computer disk, the spherical wave coefficients of the expansion of the incident fields. As mentioned previously, these coefficients are written on the disk by the SWE program.

FIELDS contains two entry points, ENTRY FIELD1 and ENTRY FIELD2. FIELD1 evaluates the SWE coefficients to find the near field form of the pattern incident on the reflector.

FIELD2 determines the electric field at an infinite distance from its source. FIELD2 is used in calculation of the far field form of the incident pattern.

Both FIELD1 and FIELD2 use the SWE coefficients in their computations.

Input variables:

TITLE - alphanumeric statement, $\leq$ 70 characters.

LMAX - maximum mode order, $\leq$ 70.

MCOMP - order of azimuthal variation.

A(N,1), A(N,2) - real and imaginary components of $TE_{MCOMP,N}$ (N = 1 to LMAX) spherical wave coefficient.

B(N,1), B(N,2) - real and imaginary components of $TM_{MCOMP,N}$ spherical wave coefficient.

Input considerations:

The user need not include the above information with the other data for the SCAT program. Instead, FIELDS reads this data directly off the computer disk. The first three variables are inputs of the SWE program; the coefficients are calculated by it. The input considerations and table of formats for these variables are given in the SWE program description.

4. SURF (PMN SSURF).

Subroutine description:

For each integration grid, this subroutine reads in all $\rho(\theta, \phi)$ values and calculates $\frac{\partial\rho}{\partial\theta}(\theta, \phi)$ and $\frac{\partial\rho}{\partial\phi}(\theta, \phi)$. There are four input cases that SURF considers; these are determined by the integer input parameter ISURF. The four cases are the following: ρ is a function of θ and ϕ and is read in from a table of values; ρ is a function of θ and is read in from a table of values; ρ is known in analytic forms as a function of θ and/or ϕ; ρ, $\frac{\partial\rho}{\partial\theta}$ and $\frac{\partial\rho}{\partial\phi}$ are all known in analytic form. In the first situation, MMAX x NMAX values are read in (every point on the integration grid); in the second case, MMAX values are read in (every θ point on integration grid); in the remaining cases, no values are read in. When $\frac{\partial\rho}{\partial\theta}$ and $\frac{\partial\rho}{\partial\phi}$ are not analytically specified, they are numerically determined using the techniques of forward and backward differences. (See Computer Methods for Science and Engineering, LaFara.) This method of computing derivatives at the points of a table of data has been found to often be more accurate than is necessary for the SCAT program. For differentiable surfaces and not overly varying integration grids (as outlined in SETUP), this accuracy is maintained down to tables of four values. Thus, MMAX and NMAX should be greater than or equal to four.

To compute ρ and/or its partial derivatives, equations in θ and/or ϕ are inserted into SURF. The values of sin θ, cos θ, sin ϕ, cos ϕ, θ and ϕ (radians) are stored for each integration grid in the variables SIT(M), COT(M), SIP(N), COP(N), T(M), P(N).

As previously mentioned, when ρ is input in tabular form, a value of 999.99999 indicates the reflector edge has just been passed. For a given ϕ, the remaining θ points on the integration grid must be assigned some arbitrary ρ value, which does not figure in any computations. If ρ is known for points beyond the edge, it would be slightly more accurate to include the value immediately beyond (on the integration grid) the edge value. Doing this would allow ρ at the edge to be interpolated rather than extrapolated.

Input variables:

ISURF - integer parameter that determines how ρ is obtained. There are four cases as follows:

- ISURF < 0 ($\neq -2$) - ρ, $\frac{\partial\rho}{\partial\theta}$ and $\frac{\partial\rho}{\partial\phi}$ all determined analytically. Read no values.
- ISURF = -2 - ρ determined analytically; $\frac{\partial\rho}{\partial\theta}$ and $\frac{\partial\rho}{\partial\phi}$ numerically computed (backward and forward differences). Read no values.
- ISURF = 0 - ρ is a function of θ only and is read (MMAX values) from a table.
- ISURF > 0 - ρ is a function of θ and ϕ and is read (MMAX x NMAX values) from a table.

F (M, N) - $\rho(\theta, \phi)$

FT (M,N) - $\frac{\partial\rho}{\partial\theta}(\theta, \phi)$

FP (M,N) - $\frac{\partial\rho}{\partial\phi}(\theta, \phi)$

Input considerations:

If the partial derivatives of ρ are to be determined numerically, MMAX and NMAX can at minimum be two but it is suggested that they be no less than four.

A value of ρ is required for every (θ, ϕ) or θ (depending on ISURF) on the integration grid.

For the case of reading in ρ for each (θ, ϕ) point on the integration grid, the θ DO loop is within the ϕ DO loop. Thus, for each ϕ value a new block of data is started.

The above two restrictions must be kept in mind when setting a value of ρ to 999.99999.

If at all possible, ρ should be represented by an equation. For this situation the input procedure is at its simplest.

Table of inputs and their formats:

	Card	Variable(s)	Format
Read in for each integration grid.	1	ISURF (For ISURF < 0, no values read.)	I5
	2 . . .	F(1, 1) to F(MMAX, 1) or to F(MMAX, NMAX)	7(1X, F9.5)

5. DIFF (PMN DIFF)

Subroutine description:

This subroutine is called by SURF to numerically approximate the derivative of a dependent variable with respect to the independent variable at each point in a table of data. The technique of forward and backward differences is employed; forward near the front of the table and backward elsewhere. By reaching back or forth into the table of data and taking the differences of the dependent variables along the way, the method is able to accurately compute derivatives of rapidly changing data within the table. The spacing of the independent variable $(\Delta\theta, \Delta\phi)$ must be constant throughout the table (integration grid). The table should contain a minimum of four values.

There are no input variables.

6. FINT (PMN EFINT, FINT)

This subroutine performs the integration of the product of the induced surface currents and the pathlength over each integration grid for each point on the output grid. It is here in the program where the great majority of number crunching is performed. This integration routine was developed by Ludwig specifically for the form of integral encountered in scattering problems. (The fast Fourier transform is not applicable.) Ludwig's method reduces by a factor of 16, relative to a Simpson's rule integration, the number of integration points needed.

In this integration routine the θ loop is embedded within the ϕ loop.

The reflector edge is handled in the following way: If IEDGE indicates an edge is present, then for a given azimuthal angle FINT compares each $\theta + \Delta\theta$ value with the edge value of θ (which is a function of ϕ). When the edge is encountered, the edge θ for ϕ and for $\phi + \Delta\phi$ are averaged. The contribution from this smaller (in some cases, slightly larger) ΔS is then evaluated by extrapolating or interpolating the pathlength and surface currents to the averaged edge θ. (See Figure 4.). The reflector, in effect, has a discontinuous edge.

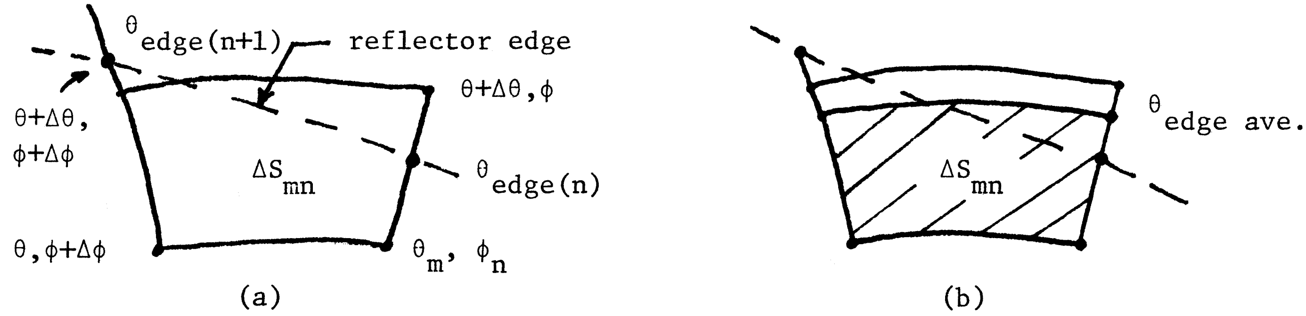


FIGURE 4

Another routine has been developed with the intent of better representing the edge. The ΔS_{mm} in Figure 4(a) is divided along ϕ, the number of divisions being determined by the difference between the two edge values of θ. θ_{edge} is then interpolated to each ϕ division. The same technique of θ averaging is applied over the reduced edge section, yielding smaller edge discontinuities. This integration routine is stored in PMN EFINT.

7. PATHL (PMN PATHL).

<u>Subroutine description:</u>

This member computes the pathlength from every point on the integration surface in the direction of each point on the output grid. The pathlength is the dot product between the vector ρ and the unit vector in the direction of the output variables.

8. SPHANK (PMN SPHANK).

<u>Subroutine description:</u>

SPHANK computes the term $h_n^{(2)}(k\rho) \cdot (-j)^{n+1} \cdot k\rho \cdot e^{jk\rho}$, where $h_n^{(2)}(k\rho)$ is the spherical Hankel function of the second kind, n is the mode order, and $j = \sqrt{-1}$.

This subroutine is called by FIELD1 (magnetic field at finite distance from source). The value SPHANK determines provides the near field ρ dependence. For far field calculations, this factor must be removed. Otherwise, SPHANK would provide only the value of $h_n^{(2)}(k\rho)$.

The spherical Hankel function represents an outward travelling spherical wave. $h_n^{(2)}(\chi)$ is defined as $j_n(\chi) - j\, y_n(\chi)$ where j_n and y_n are spherical Bessel functions. $j_n(\chi)$ and $y_n(\chi)$ are analogous to $\sin \chi$ and $\cos \chi$, and $h_n^{(2)}(\chi)$ is analogous to $e^{-j\chi}$.

SPHANK provides the only (known) numerical program limitation. For $\chi <$ 100 (roughly), $y_n(\chi)$ blows up as n increases. As χ decreases, the blow up occurs for smaller n. This is not a computing problem but is a characteristic of the spherical Bessel function, y_n. (See Handbook of Mathematical Functions, Abromowitz and Stegun; p. 438, Fig. 10.1 to 10.3 and pp. 465-466, Table 10.5.)

To avoid computer overflow SPHANK stops calculation when $y_n \sim 10^7$. For a wavelength of 5 cm ($k = 125.7\ m^{-1}$) and a plane reflector at a distance of 1 m, this truncation produced no changes in the scattered field phase or amplitude. It is suggested that the problem be such that $k\rho > 100$. For $k\rho < 100$, the above tables should be consulted to determine the maximum "safe" mode order.

There are no input variables.

9. LEGEND (PMN LEGEND)

Subroutine description:

LEGEND computes the value of the associated Legendre polynomial, P_n^m $(\cos \theta)$, where m is the order of azimuthal variation and n is the mode order.

The Legendre polynomial provides the correct polar angle (θ and Θ) variation of the spherical wave field representation. Therefore, it is called by both FIELD1 and FIELD2.

LEGEND has no numerical difficulties analogous to those of SHPANK. (See Tables of Functions, Johnke and Emde; pp. 112-113.)

There are no input variables.

10. VECTOR (PMN VECTOR)

Subroutine description:

This subroutine converts complex numbers in rectangular form (real and imaginary) to polar form (amplitude and phase). Most program computations are done with numbers in rectangular form.

There are no input variables.

11. ADJUST (PMN ADJUST)

Subroutine description:

ADJUST normalizes phase angles to the range of -180° to +180°.

There are no input variables.

12. EDGEEQ (within PMN SSURF)

Subroutine description:

Equations for the edge value of theta are inserted into this subroutine. The same values of sin ϕ, cos ϕ, etc., used in SURF are available. The edge equations will be functions of ϕ only.

If necessary, the equations can be divided up for the different ϕ integration grids.

There are no input variables.

B. Output of Scattering Program.

The printout of the program is fairly self-explanatory. Briefly, the bulk of the output is the following:

Output for each integration grid:

-- Output grid of all integration grids.

-- Spherical wave coefficients.

-- If edge is to be encountered, all edge values of theta.

-- A selection of the values of ρ, $\frac{\partial\rho}{\partial\theta}$ and $\frac{\partial\rho}{\partial\phi}$.

-- Scattered fields (E and H planes of far-electric field) from integration grid.

-- Far field incident pattern.

-- Superposition of incident fields and all grid scattered fields (i.e., yields total scattered fields).

C. Subroutines of Spherical Wave Expansion Program.

The SWE program determines the coefficients of the spherical wave expansion of an input far field pattern, such as those typically measured on an antenna range. In principle, this expansion is completely general and provides one with the near field values, including radial components, of the input pattern. (For a mathematical account of spherical wave theory, see Electromagnetic Theory, Stratton; Chapter 7.)

1. MAIN (PMN MAINOR)

Subroutine description:

This routine is essentially the entire program. It reads in the far field pattern, computes the coefficients, writes the coefficients into a disk data set, and computes the far field form of the spherical wave expansion (for comparison with the input pattern).

Input variables:

TITLE	-	Alphanumeric statement, ≤ 72 characters (identify program).
MCOMP	-	Order of azimuthal variation.
LMAX	-	Maximum mode order (θ), ≤ 80.
TITLE	-	Same as above (identify input pattern).
JIN	-	Number of input field points, ≤ 121.
IC1	-	If ≤ 0, convert incident field from dB to volts.
IC2	-	If > 0, neglect incident field phase.
IC3	-	If > 0, compute incident field amplitude from equations inserted in MAIN.
IC4	-	If > 0, $E_\theta = E_\phi$ in phase and amplitude; input each phase and amplitude once.
PSI	-	Polar angle θ.
E	-	$E_\theta(\theta, \phi)$ amplitude as a function of θ of input pattern; volts or dB.
EP	-	$E_\theta(\theta, \phi)$ phase; degrees.
H	-	$E_\phi(\theta, \phi)$ amplitude; volts or dB.
HP	-	$E_\phi(\theta, \phi)$ phase.

Input variables (continued):

JMAXO - $180/\Delta\Theta + 1$ where $\Delta\Theta$ is the desired output increment of the far field SWE pattern.

JOUT - Number of output values starting with $\Theta = 0°$.

Input considerations:

As mentioned previously, only one azimuthal expansion component can be handled by the SWE program. This component is usually m = 1, although it can be zero or > 1. The m = 1 mode typically corresponds to a pattern amplitude which varies as a full cycle sinusoid in total azimuth. It is for this case that one refers to the "E and H planes" of a pattern. If more than one azimuth component is needed, the components can be run separately and the resulting coefficients superimposed.

The number of n modes necessary to accurately represent the input pattern depends on the complexity of that pattern. For the E plane (E_θ) pattern of the C-band Cassegrain feed at 5 cm, roughly n = 70 modes were needed. Since the SWE program requires only about one minute of CPU time, the user can easily test various maximum mode orders and check which yields the most accurate far field pattern.

Comparing the far field spherical wave expansion pattern to the input pattern, one can see that the former tends to oscillate about the latter. Frequent sampling of the input pattern will help to minimize this. For the 5 cm case above, feed pattern values were input every 0.25° until the input pattern was about 35 dB down ($\theta = 20°$) and every 0.5° out to -45 dB ($\theta \approx 30°$).

An oscillation of several hundredths of a volt in the approximation pattern values is of little concern since the input pattern is usually not reliable to such resolutions. A more important consideration is that the patterns contain roughly the same power through the average angle out to the edge of the reflecting surface.

Table of inputs and their formats:

Card		Format
1	TITLE	18A4
2	MCOMP, LMAX	2I5
3	TITLE	18A4
4	J1N, IC1, IC2, IC3, IC4	5I5
5	IC4 ≤ 0 : T(1), E(1), EP(1), H(1), HP(1)	5(1X, F9.5)
	IC4 > 0 : T(1), E(1), EP(1)	3(1X, F9.5)
⋮	⋮	⋮
JIN+4	IC4 ≤ 0 : T(JIN), E(JIN), EP(JIN), H(JIN), HP(JIN)	5(1X, F9.5)
	IC4 > 0 : T(JIN), E(JIN), EP(JIN)	3(1X, F9.5)
JIN+5	JMAXO, JOUT	2I5

2. MULT (PMN MULT)

Subroutine description:

MULT multiplies two matrices. The matrix dimensions are specified in the CALL MULT statement. If any variable dimensions are changed in MAIN, the CALL MULT statement must be changed accordingly.

3. LEGEND (PMN LEGEND)

4. VECTOR (PMN VECTOR)

LEGEND and VECTOR described previously.

D. Output of the SWE Program.

-- E and H planes of input pattern in volts and degrees.

-- Real and imaginary values of TE and TM wave coefficients for each mode.

-- Fraction of total mode power of the coefficients for each mode.

-- Total mode power of coefficients.

-- Far field summation of spherical modes;
E and H planes in volts and degrees.

E. Sumitting a Job.

It will be assumed that the user is already familiar with the Pandora command system of the IBM 360. If the user is not familiar with the system, the Pandora Guide (assembled by the Charlottesville Computer Division) nicely explains it. Some essential member-oriented commands are the following: ENTER, FETCH, SAVES, CLEAR, SCRATCH, CWS, CONCAT, SEQUENCE and SUBMIT. Some essential line editing commands are the following: CHANGE, INSERT, DELETE, MOVE, COPY, EDIT and SEEK.

The necessary JCL (Job Control Language) parameters for the programs are contained in the Pandora members JCLSCAT and JCLSWE (some message suppression). Only the TIME parameter is set by the user. This specifies a maximum CPU time in minutes. In JCLSWE, TIME = 1 and need not be adjusted.

To submit the SCAT program, the following statement is entered:

SUBMIT_JCLSCAT_MAIN1SC_SSURF_FINT_SFPVSALD_DATANAME

(PMN SFPVSALD contains eight subroutines.)

In this statement, JCLSCAT must be first and SFPVSALD and DATANAME must be last and in the shown order. The three other members can be arbitrarily shuffled. DATANAME is specified by the user.

To submit the SWE program, the following statement is entered:

SUBMIT_JCLSWE_MAINOR_LVM_DATANAME

(PMN LVM contains three subroutines.)

This statement must be entered with the order as shown.

It must be kept in mind that whenever the incident field is to be changed in the SCAT program, the SWE program must be run so that the proper coefficients are stored on the disk.

The following Pandora members may prove useful:

DSCATSUB - Contains input data for scattering the 140-ft subreflector; $\Delta\theta = 0.2°$, $\Delta\phi = 2°$.

DSWE5G - Input data for SWE program. For 5 GHz C-band Cassegrain feed pattern.

DSWE10G - For 10 GHz X-band Cassegrain feed.

Shown in Figure 5 is the scattered pattern from the 140-ft subreflector at 5 GHz.

IV. SOME OF THE THEORY UNDERLYING THE PROGRAMS

A. Determining the Scattered Fields.

1. Problem definition:

Given a perfectly conducting reflecting surface and the magnetic fields incident on this surface, the far field scattered pattern is to be found. The incident electric field does not contribute to the fields scattered from a perfect conductor.

In the coordinate system of Figure 1, the following variables are defined:

$\overline{E}_s(R,\Theta,\Phi)$ = far zone scattered electric field = $[E_\Theta(\Theta,\Phi)\,\hat{\Theta} + E_\Phi(\Theta,\Phi)\,\hat{\Phi}]\,\frac{e^{-jkR}}{R}$, $R \to \infty$.

$\hat{R},\hat{\Theta},\hat{\Phi}$ = unit vectors.

$\overline{H}_i(\rho,\theta,\phi)$ = incident magnetic field.

$\hat{n}$ = outward unit normal to scattering surface.

$\hat{\rho},\hat{\theta},\hat{\phi}$ = unit vectors.

$\overline{K}(\rho,\theta,\phi)$ = induced surface current.

$\overline{\rho}$ = $\rho\,\hat{\rho}$.

$\hat{i},\hat{j},\hat{k}$ = Cartesian unit vectors.

k = $2\pi/\lambda$.

j = $\sqrt{-1}$

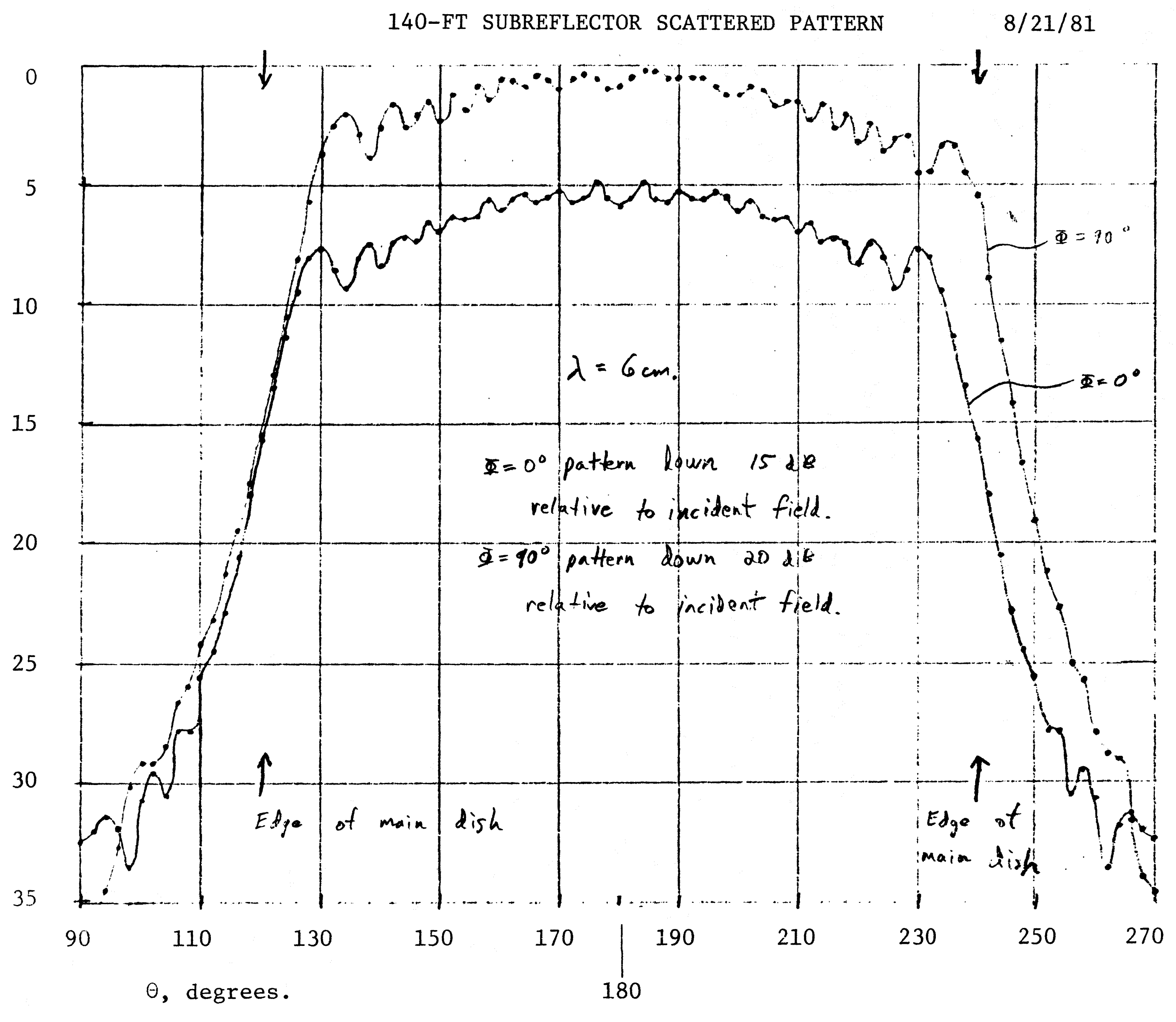

140-FT SUBREFLECTOR SCATTERED PATTERN
8/21/81
0
5
10
15
20
25
30
35
90
110
130
150
170
180
190
210
230
250
270
Θ, degrees.
Φ = 90°
Φ = 0°
λ = 6 cm.
Φ = 0° pattern down 15 dB relative to incident field.
Φ = 90° pattern down 20 dB relative to incident field.
Edge of main dish
Edge of main dish

FIGURE 5

The program computes E_Θ and E_Φ in volts. If near field values are needed, then E_Θ and E_Φ can be input to the spherical wave expansion (SWE) program.

To find the total scattered field, the far field incident pattern is added to E_s.

2. <u>Induced currents and the physical optics approximations:</u>

The scattered field, E_s, is determined by taking the 2-dimensional Fourier transform (mapping from $\theta,\phi \rightarrow \Theta,\Phi$) of the array of current dipoles. Diffraction effects will be accounted for since the wavefronts of each current dipole are summed. In principle, the accuracy of the technique is limited by the precision with which the induced currents are known.

To determine $\overline{K}$, the following assumptions are made: On directly illuminated sections of the reflector, the induced currents are those of an optically reflected $\overline{H}_i$; on shadowed portions, there are no induced currents. These two assumptions comprise the physical optics approximations.

An expression for $\overline{K}$ can be found by applying the field boundary conditions to the scattering surface. For two arbitrary mediums,

$$\overline{K} = \hat{n} \times (\overline{H}_1 - \overline{H}_2).$$

Since there are no fields beneath the reflector surface, $\overline{H}_2 = 0$ and since $\overline{H}_i$ is optically reflected, $\overline{H}_1 = 2\overline{H}_i$. Thus

$$\overline{K} = 2\hat{n} \times \overline{H}_i \qquad (1)$$

Note that $\overline{K}$ gives rise to all fields (incident and scattered) in free space (medium 1).

Ludwig notes that this expression is, in some cases, a poor approximation to the true induced currents. Generally, the calculated currents oscillate about the true currents, making little net contribution to the scattered fields. Equation (1) can be accurately used for reflectors as small as several wavelengths in diameter.

3. Fields due to the induced currents.

To determine the scattered field at some far field output point (Θ,Φ), the amplitude and phase delay (pathlength) of the radiation from each current dipole is vectorially summed. In principle, the current dipoles are of infinitestimal length; for the numerical integration this, of course, is not true.

For each dS, $d\overline{E}_s$ is proportional to the currents induced on dS. Thus, the radiation amplitude is $\overline{K}$.

Since the far field of $\overline{E}_s$ is desired, it is instructive to view the output grid, over which the scattered pattern is calculated, as lying on a sphere of infinite radius centered on the source phase center. If the source fields travel directly to the output grid without striking a reflector, the pathlength covered is defined to be zero. When intercepted first by a reflector, the pathlength is defined as the additional distance travelled due to the reflection. This additional distance is that covered by the source fields to the reflector minus the extent to which the incident fields have already travelled in the direction of the output point (extent of colinearity). If R and vector $\hat{\rho}$ are colinear, the pathlength is, by definition, zero. A measure of the colinearity of two vectors is the cosine of the angle between them (dot product). Thus, the pathlength is

$$\gamma = \rho(1 - \hat{\rho} \cdot \hat{R}).$$

To evaluate $\hat{\rho} \cdot \hat{R}$, define the following:

$$x = \rho \sin\theta \cos\phi \qquad X = R \sin\Theta \cos\Phi$$

$$y = \rho \sin\theta \sin\phi \qquad Y = R \sin\Theta \sin\Phi$$

$$z = \rho \cos\theta \qquad Z = R \cos\Theta$$

and

$$\hat{\rho} = (x\hat{i} + y\hat{j} + z\hat{k})/\rho, \quad \hat{R} = (X\hat{i} + Y\hat{j} + Z\hat{k})/R.$$

The dot product is then

$$\hat{\rho} \cdot \hat{R} = [\sin\theta \sin\Theta (\cos\phi \cos\Phi + \sin\phi \sin\Phi) + \cos\theta \cos\Theta].$$

The pathlength must be normalized to the wavelength. For an outward travelling spherical wave, the phase delay from source phase center to output point is

$$\text{Phase delay} = e^{-jk\gamma}.$$

At this point, our knowledge of the scattered fields can be summarized by

$$\overline{E}_s \propto \frac{e^{-jkR}}{R} \int_s \overline{K} e^{-jk\gamma} ds,$$

where $\overline{E}_s$ and $\overline{K}$ are expressed in volts/meter and amperes/meter, respectively. The variable pathlength is neglected in computing the spacial attenuation (1/R) of $\overline{E}_s$, since R goes to infinity. Therefore,

$$E_\Theta \hat{\Theta} + E_\Phi \hat{\Phi} \propto \int_s \overline{K} e^{-jk\gamma} ds. \tag{2}$$

To get the two sides of equation (2) compatible, we must convert the current-distance of the right side to voltage. Recalling the phase quadrature of associated currents and voltages, the constant of proportionality is $-2\pi j Z_0/\lambda$, where Z_0 = the impedance of free space. The usual expression for this constant is $-j\omega\mu_0$, where μ_0 is the magnetic permeability of free space and $\omega = 2\pi$ times the frequency. Define

$$\overline{I}(\Theta,\Phi) = [E_\Theta \hat{\Theta} + E_\Phi \hat{\Phi}]/-j\omega\mu_0 \quad \text{(ampere-meters)}.$$

Then

$$\overline{I} = \frac{1}{4\pi} \int_s \overline{K} e^{-jk\gamma} ds.$$

The $\frac{1}{4\pi}$ is a normalization constant which arises from integrating over a solid angle. A sphere has 4π square radians (steradians).

Since $\overline{H}_i$ will often be evaluated within the source's near field, it will have a relatively strong radial (ρ) dependence. However, this dependence is predominantly of the form $1/\rho$. (The $e^{-jk\rho}$ phase term has already been included in the expression for γ.) Define:

$$\overline{H}_i = \frac{1}{\rho} [H_\rho \hat{\rho} + H_\theta \hat{\theta} + H_\phi \hat{\phi}],$$

where H_ρ, H_θ and H_ϕ are complex values, thus allowing for phase deviating from the $e^{-jk\rho}$ form. Since H_ρ, H_θ and H_ϕ are slowly varying with ρ, the numerical integration is easier.

Finally, some results from differential geometry concerning normals to surfaces must be considered. We have defined $\overline{\rho}$ as $\overline{\rho} = \overline{\rho}(\theta,\phi)$; specifying θ and ϕ describes the surface. Figure 6 shows the vectors $\overline{\rho}$, $\frac{\partial\overline{\rho}}{\partial\phi}$ and $\frac{\partial\overline{\rho}}{\partial\theta}$. The latter two are tangent to the surface at (θ,ϕ) and are normal to each other. As such, their cross product defines a normal to the surface at (θ,ϕ). After some staring at Figure 6, one can see that $\frac{\partial\overline{\rho}}{\partial\theta}$ and $\frac{\partial\overline{\rho}}{\partial\phi}$ are the resultants of vectors in the directions $\hat{\rho}$ and $\hat{\theta}$ and the directions $\hat{\rho}$ and $\hat{\phi}$, respectively.

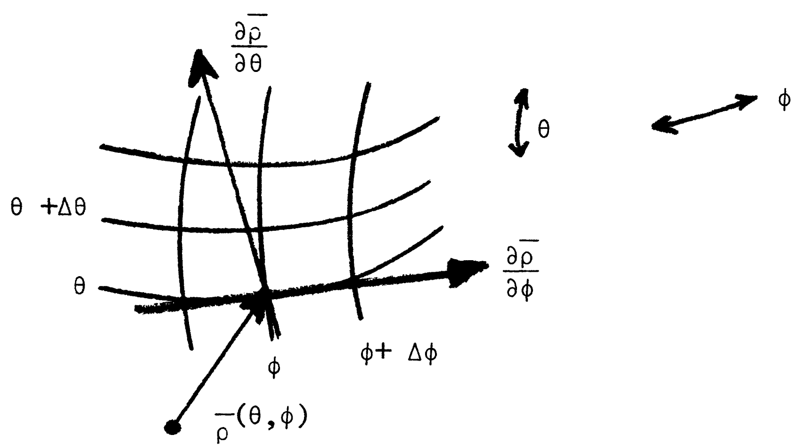


FIGURE 6

Evaluating these derivatives (Figure 1 is helpful), we find:

$$\frac{\partial \overline{\rho}}{\partial \theta} = \left(\frac{\partial \rho}{\partial \theta}\right) \hat{\rho} + \rho \left(\frac{\partial \hat{\rho}}{\partial \theta}\right) = \frac{\partial \rho}{\partial \theta} \hat{\rho} + \rho \hat{\theta}$$

$$\frac{\overline{\partial \rho}}{\partial \phi} = \left(\frac{\partial \rho}{\partial \phi}\right) \hat{\rho} + \rho \left(\frac{\partial \hat{\rho}}{\partial \phi}\right) = \frac{\partial \rho}{\partial \phi} \hat{\rho} + \rho \sin \theta \hat{\phi}$$

and

$$\frac{\partial \overline{\rho}}{\partial \phi} \times \frac{\partial \overline{\rho}}{\partial \theta} = \rho \frac{\partial \rho}{\partial \phi} \hat{\phi} + \rho \sin \theta \frac{\partial \rho}{\partial \theta} \hat{\theta} - \rho^2 \sin \theta \hat{\rho}.$$

The unit normal, $\hat{n}$, can be found from the relationship

$$\hat{n}\, dS = \frac{\partial \overline{\rho}}{\partial \phi} \times \frac{\partial \overline{\rho}}{\partial \theta} d\theta\, d\phi$$

where $dS = (\rho \sin \theta\, d\phi)\ (\rho d\theta) = \rho^2 \sin \theta\, d\phi\, d\theta$. Subsituting for $\hat{n}\, dS \times \overline{H}_i$ in the equation for $\overline{I}$, we obtain

$$\overline{I} = \frac{1}{2\pi} \int_s \overline{F}(\theta,\phi)\, e^{-jk\gamma}\, d\theta\, d\phi$$

where

$$\overline{F} = \left\{\frac{\partial \rho}{\partial \theta} \sin \theta\, H_\phi - \frac{\partial \rho}{\partial \phi} H_\theta\right\} \hat{\rho} + \left\{\frac{\partial \rho}{\partial \phi} H_\rho + \rho \sin \theta\, H_\phi\right\} \hat{\theta}$$

$$+ \left\{- \rho \sin \theta\, H_\theta - \sin \theta \frac{\partial \rho}{\partial \theta} H_\rho\right\} \hat{\phi} \quad .$$

This is the form in which the radiation integral is evaluated.

B. Technique of Numerical Integration.

1. Form of integral:

Dropping all constants, the form of the integral over a given integration grid and for the output point Θ_J, Φ_K is

$$I(\Theta_J, \Phi_K) = \int_{\theta_1}^{\theta_M} \int_{\phi_1}^{\phi_N} F(\theta,\phi) \exp[jk\gamma\ (\theta, \phi, \Theta_J, \Phi_K)] d\theta\ d\phi$$

Define $\Delta\Omega_{mn}$ as the piece of solid angle bounded at its four corners by θ_m, θ_{m+1}, ϕ_m, and ϕ_{m+1}. Also, define

$$\Delta\theta_m = \theta_{m+1} - \theta_m$$

$$\Delta\phi_n = \phi_{n+1} - \phi_n$$

$$\rho_{mn},\ F_{mn},\ \text{etc.} = \rho(\theta_m, \phi_n),\ F(\theta_m, \phi_n),\ \text{etc.}$$

The strong sinusoidal variation of the phase delay creates the need for rapid sampling of the scattering surface. Consider a $\Delta\Omega_{mn}$ with physical dimensions on the order of a wavelength; $\rho_{mn}\Delta\theta_m \sim \lambda$ and $\rho_{mn} \sin \theta_m\ \Delta\phi_n \sim \lambda$. (The dimension of one wavelength is chosen because electromagnetic fields rarely change abruptly over such distances.) For this situation the phase delay can vary up to one full cycle as $\Delta\Omega_{mn}$ is traversed. To successfully apply an integration technique such as Simpson's rule would require further subdivision of the scattering surface, thus creating a monstrous CPU time. Note that because γ is a nonlinear function of θ and ϕ, the fast Fourier transform cannot be applied.

2. <u>The integration technique:</u>

The sampling hassle can be alleviated through use of a linear representation of F and γ (especially) over each $\Delta\Omega_{mn}$. (The procedure to be described below was developed by Ludwig and is shown on page 13 of his JPL report.) Approximate F and γ by

$$F(\theta,\phi) = a_{mn} + b_{mn}\ (\theta - \theta_m) + c_{mn}\ (\phi - \phi_n)$$

$$\gamma(\theta,\phi) = \alpha_{mn} + \beta_{mn}\ (\theta - \theta_m) + \xi_{mn}\ (\phi - \phi_n)$$

where θ and ϕ assume the corner values of $\Delta\Omega_{mn}$. Applying a least squares plane fit to F and γ at the corners of $\Delta\Omega_{mn}$, the following normal equations are obtained for F:

$$\Sigma F = a_{mn}\, \Sigma\, 1 + b_{mn}\, \Sigma\, (\theta - \theta_m) + c_{mn}\, \Sigma\, (\phi - \phi_n)$$

$$\Sigma F(\theta - \theta_m) = a_{mn}\, \Sigma\, (\theta - \theta_m) + b_{mn}\, \Sigma\, (\theta - \theta_m)^2 + c_{mn}\, \Sigma\, (\phi - \phi_n)\,(\theta - \theta_m)$$

$$\Sigma F(\phi - \phi_n) = a_{mn}\, \Sigma\, (\phi - \phi_n) + b_{mn}\, \Sigma\, (\theta - \theta_m)\,(\phi - \phi_n) + c_{mn}\, \Sigma\, (\phi - \phi_n)^2$$

where
$$\Sigma = \sum_{n}^{n+1} \sum_{m}^{m+1}$$

Solving for the coefficients yields the following:

$$a_{mn} = \frac{1}{4}\,[3\,F_{mn} - F_{m+1\ n+1} + F_{m+1\ n} + F_{m\ n+1}]$$

$$b_{mn} = \frac{1}{2\Delta\theta_m}\,[F_{m+1\ n} - F_{mn} + F_{m+1\ n+1} - F_{m\ n+1}]$$

$$c_{mn} = \frac{1}{2\Delta\phi_n}\,[F_{m\ n+1} - F_{mn} + F_{m+1\ n+1} - F_{m+1\ n}]$$

The results are the same for γ and its coefficients.

Substituting the approximations for F and γ into equation (2) and integrating over $\Delta\Omega_{mn}$, the scattered field contribution ΔI_{mn} can analytically be determined. The expression for ΔI_{mn} is

$$\Delta I_{mn} = \exp jk\alpha_{mn} \left\{ a_{mn} \left[\frac{e_m - 1}{jk\beta_{mn}}\right] \left[\frac{e_n - 1}{jk\xi_{mn}}\right] \right.$$

$$+ \; b_{mn} \left[\frac{\Delta\theta_m}{jk\beta_{mn}} \; e_m - \left(\frac{e_m - 1}{(jk\beta_{mn})^2}\right)\right] \left[\frac{e_n - 1}{jk\xi_{mn}}\right]$$

$$\left. + \; c_{mn} \left[\frac{e_m - 1}{jk\beta_{mn}}\right] \left[\frac{\Delta\phi_n}{jk\xi_{mn}} \; e_n - \left(\frac{e_n - 1}{(jk\xi_{mn})^2}\right)\right] \right\}$$

where

$$e_m = \exp jk\beta_{mn} \, \Delta\theta_m$$

$$e_n = \exp jk\xi_{mn} \, \Delta\theta_n$$

For each θ_J, Φ_K of the output grid, the integration subroutine (FINT) computes I from all θ_m, ϕ_n on the integration grid. At every integration point the subroutine computes the least squares coefficients and evaluates the above expression for ΔI_{mn}. To avoid numerical catastrophe an altered expression of ΔI_{mn} is used for β_{mn} and/or ξ_{mn} near zero.

3. Some comments on the method:

Analytic evaluation of the radiation integral can be thought of as summing the (relatively simple) patterns due to infinitesimal current dipoles. The numerical technique sums the patterns, given by ΔI_{mn}, for surface elements a wavelength on a side.

Ludwig claims that for scattering from a hyperpoloid, a $\Delta\Omega_{mn}$ 2/3 square wavelengths in size results in errors 40 dB below the scattered pattern maximum.

C. A Few Details of Spherical Wave Expansions.

1. The SWE representation of an electromagnetic field

Define the following variables:

$\overline{E}$, $\overline{H}(\rho,\theta,\phi)$	=	description everywhere in a source-free region V of an electromagnetic field.
TE_{mn}, TM_{mn}	=	transverse electric and transverse magnetic fields used to describe $\overline{E}$ and $\overline{H}$.
m	=	order of azimuthal variation of TE, TM fields.
n	=	mode order of TE, TM fields.
$\overline{m}$, $\overline{n}$	=	spherical wave solutions to Maxwell's equations; define TE and TM fields.
a_{mn}, b_{mn}	=	TE and TM expansion coefficients.
$z_n(k\rho)$	=	any solution to spherical Bessel (differential) equation.
$h_n^{(2)}(k\rho)$	=	spherical Hankel function (a particular z_n).
$P_n^m(\cos\theta)$	=	associated Legendre function (solution to a form of the Legendre differential equation).

The SWE is used (in this program) to represent a far field input pattern, $\overline{E}$ and $\overline{H}$, in both the near and far fields. An expansion in TE and TM spherical waves is used as shown:

$$\overline{E} \text{ or } \overline{H} = \sum_m \sum_n a_{mn} \, TE_{mn} + b_{mn} \, TM_{mn} .$$

For $\overline{E}$, TE_{mn} has no radial components and for $\overline{H}$, TM_{mn} has no radial components.

As discussed earlier, the SWE program calculates the complex-valued wave coefficients. Input for $\overline{H}$ is the far-zone magnetic field of, for example, a feed; $\overline{E} = 0$. The program reduces the flexibility of the expansion in two ways: one, by requiring that m assume a single integer value (usually m = 1) and eliminating the summation over m; and, two, by discarding the odd solutions of $\overline{m}_{mn}$ and $\overline{n}_{mn}$, thus requiring that $\overline{H}$ be linearly polarized. (The existence of even and odd solutions has been, for convenience, left unrecognized in the notation.)

The general spherical wave expansion is as follows:

Let a sphere of radius ρ_0 contain all field sources. Then the electromagnetic field in the (source-free) region V that includes all space outside of the sphere of radius ρ_0 is

$$\overline{E}(\rho,\theta,\phi) = -\sum_m \sum_n a_{mn} \overline{m}_{mn} + b_{mn} \overline{n}_{mn}$$

$$\overline{H}(\rho,\theta,\phi) = \frac{k}{j\omega\mu} \sum \sum a_{mn} \overline{n}_{mn} + b_{mn} \overline{m}_{mn}$$

where $\omega\mu/k$ is Z_0 = free space impedance and where

$$\overline{m}_{mn} = \mp z_n(k\rho) \frac{m P_n^m (\cos\theta)}{\sin\theta} \begin{matrix}\sin\\ \cos\end{matrix} m\phi \, \hat{\theta}$$

$$- z_n(k\rho) \frac{\partial}{\partial\theta} P_n^m (\cos\theta) \begin{matrix}\cos\\ \sin\end{matrix} m\phi \, \hat{\phi}$$

$$\overline{n}_{mn} = n(n+1) \frac{z_n(k\rho)}{k\rho} P_n^m (\cos\theta) \begin{matrix}\sin\\ \cos\end{matrix} m\phi \, \hat{\rho}$$

$$+ \frac{1}{k\rho} \frac{\partial}{\partial\rho} [\rho z_n(k\rho)] \frac{\partial}{\partial\theta} P_n^m (\cos\theta) \begin{matrix}\sin\\ \cos\end{matrix} m\phi \, \hat{\theta}$$

$$\pm \frac{1}{k\rho} \frac{\partial}{\partial\rho} [\rho z_n(k\rho)] \frac{m P_n^m (\cos\theta)}{\sin\theta} \begin{matrix}\cos\\ \sin\end{matrix} m\phi \, \hat{\phi}$$

(The even solutions contain the upper of the two signs and the upper of the sin-cos $m\phi$ pair. Further discussion of the specifics of the SWE program expansion will be delayed until section C.)

Since we are dealing with travelling spherical waves, $z_n(k\rho)$ should describe such. The spherical Bessel function that describes the radial variation of an outward travelling spherical wave is $h_n^{(2)}(k\rho)$, the Hankel function of the second kind.

Note the roles played by the mode indices, m and n; two indices describe variation in three (coordinate) variables. Theta pattern variation, as described by $P_n^m(\cos\theta)$ is dependent on both mode orders (m is not an exponent).

2. Spherical waves as solutions to Maxwell's equations:

In a source free medium, the curl relationships of Maxwell's equations completely describe an electromagnetic field, $\overline{E}$ and $\overline{H}$. That is,

$$\overline{\nabla} \times \overline{E} = -\mu_0 \frac{\partial \overline{H}}{\partial t} = -j\omega\mu_0\overline{H}$$

$$\overline{\nabla} \times \overline{H} = \varepsilon_0 \frac{\partial \overline{E}}{\partial t} = j\omega\varepsilon_0\overline{E}$$

where
$$\overline{\nabla} = \frac{\partial}{\partial\rho}\hat{\rho} + \frac{1}{\rho}\frac{\partial}{\partial\theta}\hat{\theta} + \frac{1}{\rho\sin\theta}\frac{\partial}{\partial\theta}\hat{\phi}, \quad \text{and}$$

where an $e^{j\omega t}$ time variation has been assumed and where $\omega\mu_0 = kZ_0$ and $\omega\varepsilon_0 = k/Z_0$. These equations state that the change with distance in directions perpendicular to $\overline{E}$ ($\overline{H}$) is proportional to the time rate of change of $\overline{H}$ ($\overline{E}$). The constants of proportionality, μ_0 and ε_0, specify the extent per unit distance to which free space can transmit energy in magnetic and electric fields, respectively. (μ_0 and ε_0 are, as shown above, directly related to the impedance of free space, Z_0.) The negative sign in the first curl equation is a result of the orientation of $\overline{E}$ and $\overline{H}$ in a right-hand coordinate system.

Another interpretation of the curl equations is that the existence of an electric field ensures the existence of a corresponding magnetic field, and vice versa, for non-static fields.

Eliminating each variable of the curl equations yields the following:

$$\overline{\nabla} \times (\overline{\nabla} \times H) = k^2\overline{H} \qquad (1)$$

$$\overline{\nabla} \times (\overline{\nabla} \times \overline{E}) = k^2\overline{E}$$

Using vector identities in rectangular coordinates

$$(\overline{\nabla} = \frac{\partial}{\partial x}\hat{i} + \frac{\partial}{\partial y}\hat{j} + \frac{\partial}{\partial z}\hat{k})$$

and noting that the divergence of $\overline{E}$ and $\overline{H}$ are zero ($\overline{\nabla} \cdot \overline{E}$, $\overline{\nabla} \cdot \overline{H} = 0$ in region V since no sources exist there), these equations are transformed to

$$\overline{\nabla}^2\,\overline{H} + k^2\,\overline{H} = 0 \qquad \text{(Same for } \overline{E}.) \qquad (2)$$

This is the vector wave or vector Helmholtz equation.

∇^2 is often called the Laplacian. The (negative of the) Laplacian of a function is closely related to the difference between the value of the function at some point and the average value of the function at straddling points. That is, $\overline{\nabla}^2$ determines the concavity or lumpiness of a function. Thus, equation (2) states that the concavity of the magnitude or direction of $\overline{H}$ is proportional to $\overline{H}$.

Since the curl of a vector function yields the magnitude and direction of the rotation, or vorticity, of the vector field, equation (1) can be understood as requiring that the vorticity lines of $\overline{E}$ and $\overline{H}$ must themselves exhibit vorticity. Further, this second order vorticity is proportional in magnitude and direction to E and H. (See Methods of Theoretical Physics, Morse and Feshback; chapter 1.) Note that in a source-free space, equations (1) and (2) are identical.

The vector solutions $\overline{m}_{mn}$ and $\overline{n}_{mn}$ of equation (2) are most easily found by considering some physical attributes of $\overline{E}$ and $\overline{H}$. Recall that the divergence of $\overline{E}$ and $\overline{H}$ is zero, allowing them to be represented by the curl of a vector potential $\overline{A}$. (To this can be added the gradient of a scalar function. However, since the curl of the gradient of a function is identically zero, this term is of no use.) If $\overline{H} = \frac{1}{\mu_0}$ curl $\overline{A}$, then from Maxwell's curl equations,

$$\overline{E} = -\frac{\mu_0}{Z_0}\frac{\partial\overline{A}}{\partial t} = -jk\overline{A}.$$

Our choice of $\overline{A}$ can greatly reduce the complexity of the problem. To ease the pain of applying boundary conditions in spherical coordinates, the direction of $\overline{A}$ is (if possible) chosen to be normal to a boundary surface. This will create fields tangent to the surface boundary. The magnitude of $\overline{A}$ is $\psi(\rho,\theta,\phi)$, a rectangular component of $\overline{E}$ or $\overline{H}$, times a coordinate scale factor. For spherical coordinates the direction and magnitude of $\overline{A}$ are $\hat{\rho}$ and $\rho\psi$, respectively. Thus, a solution to equation (2) is

$$\overline{H} = \overline{\nabla} \times \overline{\rho\psi} = \overline{m} \tag{3}$$

where $\overline{\nabla}$ is in spherical coordinates. Then, equation (3) has no radial components and, if ψ is a component of $\overline{H}$, $\overline{m}$ is a TM field.

We must, of course, have radial electromagnetic field components in a general field representation. Another solution to (2) which will supply the needed components is simply the curl of $\overline{m}$. Including a $\frac{1}{k}$ constant of proportionality, a second solution to (2) is

$$\overline{H} = \frac{1}{k}\overline{\nabla} \times \overline{m} = \overline{n}.$$

In general, $\hat{n}$ has ρ, θ and ϕ components. (For detailed mathematical accounts of the solution to (2), see Stratton, chapter 7, and Morse and Feshback, chapter 13.)

The problem has been reduced to determining ψ. Recall that ψ is a solution to

$$\overline{\nabla}^2 H_p, E_p + k^2 H_p, E_p = 0 \quad , \quad p = x, y, z.$$

where E_p and H_p are functions of ρ, θ and ϕ. $\overline{\nabla}^2$ can then be expressed in spherical coordinates. The resulting differential equation is evaluated by the separation of variables

$$\psi(\rho,\theta,\phi) = \psi_1(\rho)\ \psi_2(\theta)\ \psi_3(\phi).$$

The solutions to the three resulting ordinary differential equations are:

$$\psi_1(\rho) = z_n(k\rho) = h_n^{(2)}(k\rho)$$

$$\psi_2(\theta) = P_n^m\ (\cos\theta)$$

$$\psi_3(\phi) = \begin{matrix}\cos\\ \sin\end{matrix}\ m\phi \quad \text{(See Stratton, chapter 7.)}$$

The previously shown form of $\overline{m}$ and $\overline{n}$ can be found from curl $\psi\hat{\rho}$ and $\frac{1}{k}$ curl curl $\psi\overline{\rho}$.

3. <u>What are spherical waves?</u>

To understand, physically, what spherical waves are, it will be helpful to be aware of some mathematical characteristics of the spherical Hankel function and the associated Legendre function.

The spherical Hankel function of the second kind is defined as

$$h_n^{(2)}(\chi) = j_n(\chi) - j\, Y_n(\chi), \; j = \sqrt{-1}$$

At $\chi = 0$, j_n is finite and Y_n has a pole. As $\chi_1 = k\rho_1 \to \infty$

$$h_n^{(2)}(\chi_1) = j^{n+1} \frac{e^{-j\chi_1}}{\chi_1}$$

and

$$\frac{1}{\chi_1} \frac{\partial}{\partial \chi} \left[\chi h_n^{(2)} (\chi) \right]_{\chi=\chi_1} = j^n \frac{e^{-j\chi_1}}{\chi_1}$$

These are closely related to the $e^{-jk\rho}/\rho$ dependence of a far field pattern.

The associated Legendre function provides the polar angle field variation. The order of index n runs from 0 to ∞, and m runs 0 to n. A few modes of the function are shown below. (See Stratton, chapter 7; Tables of Function, Johnke and Emde, pp. 112-113.)

$$P_0(Z) = 1$$

$$P_1(Z) = Z = \cos\theta$$

$$P_1^1(Z) = (1 - Z^2)^{1/2} = \sin\theta$$

$$P_2(Z) = \frac{1}{2}(3Z^2 - 1) = \frac{1}{4}(3\cos 2\theta + 1)$$

$$P_2^1(Z) = 3(1 - Z^2)^{1/2} Z = \frac{3}{2}\sin 2\theta$$

$$P_2^2(Z) = 3(1 - Z^2) = \frac{3}{2}(1 - \cos 2\theta)$$

The functions $\cos m\phi\, P_n^m(\cos\theta)$ and $\sin m\phi\, P_n^m(\cos\theta)$ are periodic on the surface of a unit sphere. For $m > 0$, the functions are zero at the poles. The number of nodal lines parallel to the equator is $n - m$. These are orthogonally intersected by the 2 m longitudinal nodes. Since the surface is divided into rectangular sections within which the above functions are alternately positive and negative, these functions are referred to as tesseral harmonics of n^{th} degree and m^{th} order.

Evaluating $\overline{m}$ and $\overline{n}$ at m = o and n = 1, the simplest form of $\overline{m}$ and $\overline{n}$, will provide the vital clue in determining the nature of these spherical wave solutions.

Recall that $\overline{E}$ and $\overline{H}$ are defined everywhere in space V which surrounds but does not include a sphere of sources. Let this sphere be of radius a. An entirely equivalent representation of the sources is to place surface currents on the sphere and remove all sources within the sphere. From the boundary conditions, the surface currents are

$$\overline{K} = \hat{\rho} \times \overline{H}(a,\theta,\phi).$$

From the expressions for $\overline{E}$ and $\overline{H}$ involving the vector potential $\overline{A}$,

$$\overline{E} = -jk\overline{A} \quad (\text{for } e^{j\omega t} \text{ time variation})$$

$$\overline{H} = \frac{1}{\mu_0} \text{curl } \overline{A}$$

where $\overline{A} = \overline{m}$ and $\overline{n}$, we can determine the induced currents. For m = 0, n = 1,

$$\overline{m}_{01} = h_1^{(2)} (k\rho) \sin\theta \, \hat{\phi}$$

$$\overline{n}_{01} = \frac{1}{k} \overline{\nabla} \times \overline{m}_{01}$$

$$= 2 \cos\theta \frac{1}{k\rho} h_1(k\rho)\, \hat{\rho} - \sin\theta \frac{1}{k\rho} \frac{\partial}{\partial\rho} [\rho h_1(k\rho)]\, \hat{\theta}.$$

First consider the case of $\overline{A} = \overline{m}_{01}$. Through the use of various recurrence relations (see Stratton, chapter 7) and the relation $h_n(Z) = -j(-1)^n \left(\frac{d}{ZdZ}\right)^n \left(\frac{e^{iZ}}{Z}\right)$, the terms involving Hankel functions can be reduced to algebraic expressions in $k\rho$ and $e^{jk\rho}$. Then the surface current creating the m = 0, n = 1 mode fields is

$$\overline{K}_{01} \propto j \frac{\omega}{c} \sin\theta \left[\frac{1 - (ka)^2 - jka}{(ka)^3}\right] e^{jka} \hat{\phi}$$

(See Morse and Feshback, chapter 13, p. 1867.)

Thus, we have a current oscillating in time parallel to the equator. The current goes to maximum at the equator and goes to zero at the poles and travels in the same direction at all points on the sphere. There is no charge build up as can be seen from the lack of radial $\overline{E}$ components.

A current of such form characterizes the magnetic dipole and $\overline{m}_{01}$ represents the field from the dipole.

If we are to determine the radial components of $\overline{E}$, we must turn the problem around and set $\overline{A} = \overline{n}_{01}$. Then, $\overline{E} = -jk\,\overline{n}_{01}$ and has a radial component. The surface current is

$$\overline{K}_{01} \propto \omega\mu_0 \sin\theta \left[\frac{j + ka}{(ka)^2}\right] e^{jka} \hat{\theta}$$

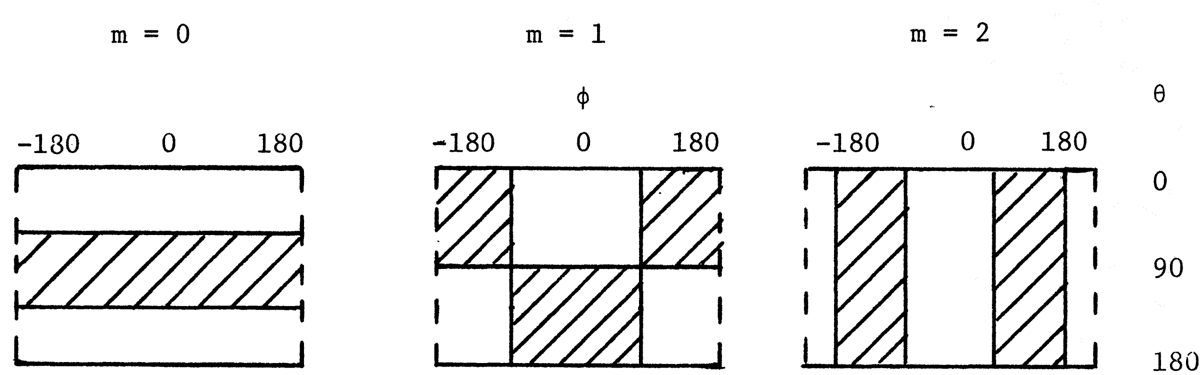
which oscillates between poles, alternately depositing positive and negative charges at the poles. As can be seen from $\hat{\rho} \cdot \overline{E}$, the charge is concentrated at the poles and is 90° out of phase with the current.

This oscillation of charge is merely a way of describing an electric dipole where $\overline{n}_{01}$ represents the field from this dipole.

For m = 0, n = 2, the TE radiation is that of a magnetic quadrupole. The current circulates parallel to the equator going clockwise in one hemisphere and counterclockwise in the other for half a cycle, then reversing for the next half cycle.

The corresponding TM radiation is that of an electric quadrupole. The current distribution describes charge moving alternately from the equator to the poles and then from the poles to the equator. Thus, free charge of one sign is accumulated at the poles and of the opposite sign at the equator, the signs changing at the next half cycle.

Among others, one important question that remains is the following: What is physically happening as m changes for a given n? Plotting $\cos m\phi \, P_n^m(\cos\theta)$ on the unit sphere for various values of m seems to indicate that m determines two properties of the multipoles: the distribution of multipoles within the array and the orientation of the multipole array. As an example, consider the unit sphere plots for n = 2 and m = 0, 1, 2:



$\frac{1}{4}(3\cos 2\theta + 1)$ $\qquad$ $\frac{3}{2}\sin 2\theta \cos\phi$ $\qquad$ $\frac{3}{2}(1 - \cos 2\theta)\cos 2\phi$

The shaded portions indicate negative value; nodal lines are solid. For m = 0, there is an overlap of 2 like (in sign) "monopoles". For m = 1 and 2, all "monopoles" (2 of each sign) are distinct but the quadrupole orientations are different.

4. Determining the SWE coefficients.

As mentioned previously, the summation over m in the spherical wave expansion is eliminated; usually an order of azimuthal variation of m = 1 is needed. The m = 1 corresponds to the situation in which a pattern may be divided into E and H planes. Since only the incident magnetic field contributes to the induced surface currents, only $\overline{H}_i$ need be expanded. The expansion is:

$$\overline{H}_i(\rho,\theta,\phi) = \frac{k}{j\omega\mu} \sum_{n=1}^{N} a_n \overline{n}_n + b_n \overline{m}_n$$

where $\overline{n}_n \equiv \overline{n}_{mn}$ and is TE and $\overline{m}_n \equiv \overline{m}_{mn}$ and is TM. This expansion will determine $\overline{H}_i$, where $\overline{H}_i$ is originally a far field pattern, anywhere in the space outside a sphere of radius ρ_0 which encloses the sources of $\overline{H}_i$.

Ludwig derives expressions for the expansion coefficients for the case in which the involved data are the tangential components of $\overline{E}$ on a sphere of radius $\rho_1 > \rho_0$. The derivation will be outlined only; the mathematical details may be found in the JPL report.

The vector character of the expansion is eliminated (momentarily) by equating the tangential field components, $E_\theta(\theta,\phi)$ and $E_\phi(\theta,\phi)$, to the summation of the corresponding components of the expansion. The azimuthal terms of the summation are removed by taking the ordinary Fourier expansion of E_θ and E_ϕ, leaving a summation independent of ϕ. $E_\theta(\theta, 90°)$ and $E_\phi(\theta, 0°)$ are the usual far field patterns being expanded. The actual inputs to the program are $A_m(\theta)$ and $B_m(\theta)$, which represent the m^{th} Fourier component of the input pattern. (For m = 1, $A_m(\theta)$ and $B_m(\theta)$ are simply the pattern values.)

Using the integral (orthogonality) properties of Legendre functions, an integral (over θ) expression is found for the coefficients (JPL report; page 23, equation 9). The far field value of the Hankel function and its derivative with respect to ρ is incorporated into the coefficients; that is, the coefficients actually computed are:

$$a_n' = a_n\, j^{n+1}\, e^{-jk\rho_1} / k\rho_1$$

$$b_n' = b_n\, j^{n}\, e^{-jk\rho_1} / k\rho_1$$

a_n' and b_n' are written on the computer disk where the SCAT program can gain access to them.

V. ACKNOWLEDGEMENTS

I would like to thank Rick Fisher for his endless helpful suggestions and comments and for his ability to keep a straight face in the midst of sometimes ridiculous questions. I would also like to thank Pat Crane and Marc Damashek for their many valuable programming hints, and Carolyn Dunkle for typing this (initially) illegible report.

VI. REFERENCES

A. Ludwig, "Calculation of Scattered Patterns from Asymmetrical Reflectors", JPL Technical Report 32-1430; 2/15/70 (CV Library).

P. Morse and H. Feshback, <u>Methods of Theoretical Physics</u>, New York: Mc-Graw-Hill Book Co. (1953).

J. A. Stratton, <u>Electromagnetic Theory</u>, 1st ed., New York and London: McGraw-Hill Book Co. (1973).

R. L. LaFara, <u>Computer Methods for Science and Engineering</u>, Rochelle Park, NJ: Hayden Book Co. (1973).

M. Abromowitz and I. Stegun, <u>Handbook of Mathematical Functions</u>, Washington: U.S. Government Printing Office (1964).

E. Jahnke and F. Emde, <u>Tables of Functions</u>, 4th ed., New York: Dover Publications (1945).

(All but Ludwig's report are available in the Green Bank Library.)

VII.

PROGRAM CODE FOR BOTH PROGRAMS

```
C        MAIN PROGRAM FOR SCATTERING FROM  ASYMMETRICAL SUBREFLECTORS
C          ASSUME SMOOTH PERFECTLY CONDUCTING SURFACE OF ARBITRARY
C          SHAPE
C
C        PROGRAM ORIGINALLY DUE TO A. LUDWIG ,FEB. 1970
CCCCCCCC
CCCCCCCC
      COMMON/GRID1/SIT( 36),COT( 36),SIP( 91),COP( 91),T(36),P( 91)
      COMMON/GRID2/SITT(181),COTT(181),SIPP(5),COPP(5),TT(181),PP(5)
      DIMENSION TITLE(18)
      DIMENSION F(36,91),FT(36,91),FP(36,91),GAM(36,91),TEDGE(91)
      INTEGER TITLE
      COMPLEX HR(36, 91),HT(36, 91),HP(36, 91)
      COMPLEX ETT( 181,5),EPP( 181,5)
      COMPLEX ETTO,EPPO
      COMPLEX A(36,91,3)
      COMPLEX STOT(3)
      COMPLEX T1,T2,T3
      COMPLEX TET,TEP
      EQUIVALENCE (A(1,1,1),HR),(A(1,1,2),HT),(A(1,1,3),HP)
      EQUIVALENCE (GAM,FT)
C
C     T,P = THETA & PHI OF INTEGRATION GRID
C     TT,PP = BIGTHETA & BIGPHI OF OUTPUT GRID
C     COMMON PARAMETERS = SIN,COS OF ABOVE ANGLES
C     F,FT,FP = RHO AND ITS DERIVATIVES WRT TO THETA & PHI
C     HR,HT,HP = INCIDENT MAGNETIC FIELDS COMPONENTS
C     A = VECTOR RELATED TO SURFACE CURRENTS
C     STOT = VALUE OF RADIATION INTEGRAL AS RETURNED FROM FINT
C     GAM = PATHLENGTH TERM (A FUNCTION OF BOTH GRIDS)
C     TEDGE = POLAR ANGLE VALUES WHICH SPECIFY REFLECTOR EDGE AS A
C             FUNCTION OF PHI
C     ETT,EPP = SCATTERED FIELDS VALUES ON THE OUTPUT GRID
C     JMAX,KMAX = # BIGTHETA,#BIGPHI POINTS ON OUTPUT GRID
C     MMAX,NMAX = # THETA,#PHI POINTS ON PRESENT INTEGRATION GRID
C     NG1 = # INTEG GRIDS
C     ETTO,EPPO = FIELDS VALUES AT A SPECIFIC OUTPUT GRID POINT
C
      READ(5,1001)TITLE
1     WRITE(6,2001)TITLE
C       INPUT TRANLATIONS TO REFERENCE PT. FOR PHASE PATTERN OF
C        OUTPUT DATA.
      READ(5,1002)PC,XT,YT,ZT,SCALE
      IF(SCALE)22,21,22
21    SCALE=1.0
22    CONTINUE
      WRITE(6,2003)PC
1003   FORMAT(5I5)
C
C       INPUT GRID DATA AND ESTABLISH OUTPUT GRID(#2) AND
C        FIRST INTEGRATION GRID(#1)
      CALL SETUP(NG1,I,JMAX,KMAX,MMAX,NMAX)
      WRITE(6,2002)TITLE
C
C       INPUT INCIDENT FIELD DATA
      CALL FIELDS
C
C       BEGIN LOOP FOR INTEGRATION GRID SEGMENTS
```

```
      DO 90 J=1,JMAX
      DO 90 K=1,KMAX
      ETT(J,K)=(0.0,0.0)
90    EPP(J,K)=(0.0,0.0)
100   WRITE(6,2002)TITLE
C
C       INPUT IEDGE
C
      READ(5,1003)IEDGE
      IF(IEDGE .NE. 11)WRITE(6,2021)
2021  FORMAT('0','    WILL ENCOUNTER REFLECTOR EDGE ON THIS INTEGRATION'
     &,' GRID SINCE IEDGE NOT= 11',/)
C
C        INITIALIZE TEDGE
C
      IF(I .NE. 1)GO TO 23
       DO 24 N=1,NMAX
24      TEDGE(N)=0.
23    CONTINUE
C
C        CHECK IF REFLECTOR EDGE IS WITHIN PRESENT INTEG GRID
C
      IF(IEDGE .EQ. 11)GO TO 28
       WRITE(6,2022)
2022   FORMAT('0','  N    THETA EDGE (DEG.)')
C
C        IF EDGE IS PRESENT,DETERMINE TEDGE. OBTAIN TEDGE ACCORDING TO
C        VALUE OF IEDGE.
C
      IF(IEDGE)25,27,26
25    CALL EDGEEQ(NMAX  ,TEDGE)
      GO TO 29
1004  FORMAT(7(1X,F9.5))
26    READ(5,1004)(          TEDGE(N),N=1,NMAX)
      GO TO 29
27    READ(5,1004)         TEDGE(1)
      DO 30 N=2,NMAX
       TEDGE(N)=TEDGE(1)
30    CONTINUE
29    WRITE(6,2019)(JJ,TEDGE(JJ),           JJ=1,NMAX)
      DO 31 N=1,NMAX
31     TEDGE(N)=TEDGE(N)*0.01745329
2019  FORMAT(1H ,  ,              I5,2X,F11.4)
28       CONTINUE
C
C       ESTABLISH SURFACE PARAMETERS(I.E. RHO &ITS DERIVS.) ON
C        INTEGRATION GRID
      CALL SURF(I,MMAX,NMAX,F,FT,FP,IEDGE,TEDGE)
      DO 410 M=1,MMAX
      DO 410 N=1,NMAX
      F(M,N)=PC*F(M,N)
410    CONTINUE
C
C       ESTABLISH INCIDENT FIELDS ON INTEGRATION GRID BY EVALUATING
C        NEAR-FIELD OF SWE
C
      CALL FIELD1(MMAX,NMAX,HR,HT,HP,F)
C
```

```
C       COMBINE SURFACE AND FIELD DATA TO DETERMINE VECTOR A
C        A CONTAINS THREE COORDINATE COMPONENTS.
C
      DO 400 M=1,MMAX
      DO 400 N=1,NMAX
      T1=FT(M,N)*SIT(M)*HP(M,N)-FP(M,N)*HT(M,N)
      T2=FP(M,N)*HR(M,N)+F(M,N)*SIT(M)*HP(M,N)/PC
      T3=-FT(M,N)*SIT(M)  *HR(M,N)-F(M,N)*SIT(M)*HT(M,N)/PC
      A(M,N,1)=T1*SIT(M)*COP(N)+T2*COT(M)*COP(N)-T3*SIP(N)
      A(M,N,2)=T1*SIT(M)*SIP(N)+T2*COT(M)*SIP(N)+T3*COP(N)
      A(M,N,3)=T1*COT(M)-T2*SIT(M)
400   CONTINUE
      WRITE(6,2007)I
C
C       BEGIN OUTPUT GRID LOOP. FOR EACH POINT ON OUTPUT GRID DETERMINE
C        SCATTERED FIELD CONTRIBUTION FROM ENTIRE INTEG GRID.
C
      WRITE(6,2006)I
      DO 500 K=1,KMAX
      TO=PP(K)/0.017453293
      WRITE(6,2011)TO
      DO 500 J=1,JMAX
C
C      ESTABLISH PATH LENGTH PARAMETER ON INTEGRATION GRID
C
      CALL PATHL(F,J,K,MMAX,NMAX,GAM)
C
C       PERFORM INTEGRATION
C
      CALL FINT(T,P,A,GAM,MMAX-1,NMAX-1,STOT,I,J,K,PC,IFDGE,TFDGE)
C
C       ASSIGN SCATTERED FIELDS AT OUTPUT POINT
C
      ETTO=COTT(J)*(STOT(1)*COPP(K)+STOT(2)*SIPP(K))-STOT(3)
     &*SITT(J)
      EPPO=STOT(2)*COPP(K)-STOT(1)*SIPP(K)
      ETTO=-(0.0,1.0)*PC/6.2831854*ETTO
      EPPO=-(0.0,1.0)*PC/6.2831854*EPPO
      TO=TT(J)/0.017453293
      A1=REAL(ETTO)
      A2=AIMAG(ETTO)
      A3=REAL(EPPO)
      A4=AIMAG(EPPO)
C
C     CONVERT   TO POLAR COORDINATES
C
      CALL VECTOR(A1,A2,ETAMP,ETPHI)
      CALL VECTOR(A3,A4,EPAMP,EPPHI)
      WRITE(6,2012)TO,ETAMP,ETPHI,EPAMP,EPPHI
C
C        SUPERIMPOSE SCATTERED FIELD VALUE WITH VALUES CORRESPONDING
C        TO OTHER INTEG GRIDS.
C
      ETT(J,K)=ETT(J,K)+ETTO
      EPP(J,K)=EPP(J,K)+EPPO
500   CONTINUE
C
C       IF MORE INTEGRATION GRIDS REMAIN LOOP BACK
C
      IF(I-NG1)600,700,700
600   CALL RESET(I,MMAX,NMAX)
      GO TO 100
C
700   I=I+1
C
C       ESTABLISH DIRECT RADIATION ON OUTPUT GRID
      WRITE(6,2009)
      DO 750 K=1,KMAX
      TO=PP(K)/0.017453293
      WRITE(6,2011)TO
      DO 750 J=1,JMAX
C
C         EVALUATE FAR-FIELD OF SWE
C
      CALL FIELD2(J,K,ETTO,EPPO)
      TO=TT(J)/0.017453293
C     WRITE(10,2020)TO,ETTO,EPPO
      A1=REAL(ETTO)
      A2=AIMAG(ETTO)
      A3=REAL(EPPO)
      A4=AIMAG(EPPO)
C
      CALL VECTOR(A1,A2,ETAMP,ETPHI)
      CALL VECTOR(A3,A4,EPAMP,EPPHI)
      WRITE(6,2012)TO,ETAMP,ETPHI,EPAMP,EPPHI
C
C         ADD DIRECT AND SCATTERED FIELDS TO YIELD TOTAL(SCATTERED)
C         FIELDS.
C
      ETT(J,K)=ETT(J,K)+ETTO
      EPP(J,K)=EPP(J,K)+EPPO
750   CONTINUE
C
C       TRANSLATE PHASE CENTER,SCALE FIELD AMPLITUDES,AND OUTPUT
C        TOTAL FIELDS
      WRITE(6,2002)TITLE
      WRITE(6,2010)XT,YT,ZT,SCALE
      DO 760 K=1,KMAX
      TO=PP(K)/0.017453293
      WRITE(6,2011)TO
      DO 760 J=1,JMAX
      TO=TT(J)/0.017453293
      A1=REAL(ETT(J,K))
      A2=AIMAG(ETT(J,K))
      A3=REAL(EPP(J,K))
      A4=AIMAG(EPP(J,K))
C
      CALL VECTOR(A1,A2,ETAMP,ETPHI)
      CALL VECTOR(A3,A4,EPAMP,EPPHI)
      DP=XT*SITT(J)*COPP(K)+YT*SITT(J)*SIPP(K)+ZT*(COTT(J)+1.0)
      DP=DP*PC*57.29578
      ETPHI=ETPHI-DP
      EPPHI=EPPHI-DP
C
C         ADJUST PHASES TO -180,180 RANGE
C
```

```
      CALL ADJUST(ETPHI)
      CALL ADJUST(EPPHI)
      ETAMP=ETAMP*SCALE
      EPAMP=EPAMP*SCALE
      WRITE(6,2012)TC,ETAMP,ETPHI,EPAMP,EPPHI
760   CONTINUE
C
      STOP
1001  FORMAT(18A4)
1002  FORMAT(10F10.4)
2001  FORMAT(1H1,'     NRAO   SCATTERING PROGRAM',//5X,18A4)
2002  FORMAT(1H0,5X,18A4)
2003  FORMAT('0','    PROPAGATION CONSTANT=',E14.8/)
2006  FORMAT(1H0,' SCATTERED FIELDS FROM GRID',I2)
2007  FORMAT(1H0,' BEGIN INTEGRATION OVER GRID',I2)
2009  FORMAT(1H1,' DIRECT RADIATION FROM INCIDENT FIELDS')
2010  FORMAT(1H0'SUPERPOSITION OF ALL GRID SCATTERED FIELDS AND DIRECT',
     &' FIELDS',/,'  PHASECENTER TRANSLATED BY X=',F10.4,'    Y=',F10.4,
     &5H   Z=,F10.4,/,
     &'  AMPLITUDE VALUES SCALED BY FACTOR OF',E15.8)
2011  FORMAT(7H0  PHI=,F7.2,/
     &17X,7HE THETA,15X,5HE PHI,/,
     &9H    THETA,2(20H      VOLTS     PHASE))
2012  FORMAT(F9.2,F11.6,F8.2,F12.6,F8.2)
2013  FORMAT(F10.2,F10.6,F10.2,F10.6,F10.2)
      END
```

```
      SUBROUTINE SURF(I,MMAX,NMAX,F,FT,FP,IEDGE,TEDGE)
C
C
      DIMENSION F(36,91),FT(36,91),FP(36,91),TEDGE(91)
      COMMON/GRID1/SIT(36),COT(36),SIP( 91),COP( 91),T(36),P( 91)
C
C        THIS SUB. PROVIDES MAIN WITH RHO,DRHO/DTHETA AND DRHO/DPHI
C
C
C        INPUT RHOO,# INTEGRATION GRIDS,# THETA INTEG. GRIDS
C
C        SOME VARIABLES:
C
C      F,FT,FP = SURFACE PARAMETERS (RETURNED TO MAIN);F IS RHO AND
C      FT AND FP ARE DRHO/DTHETA AND DRHO/DPHI.
C      ISURF = INTEGER PARAMETER THAT SPECIFIES HOW SURFACE VALUES ARE
C      OBTAINED.
C
      READ(5,1001)ISURF
      DT=(T(2)-T(1))/.01745329
      DP=(P(2)-P(1))/.01745329
      WRITE(6,2002)I,ISURF
2002  FORMAT(1H0    'FOR INTEGRATION GRID #',I3,', ISURF =',I4,/)
1001  FORMAT(3I5)
      WRITE(6,2004)
2004  FORMAT(1H ,'  M  N        F           FT           FP')
C
C        ISURF CONSIDERS FOLLOWING CASES:
C
C          ISURF<0,NOT=-2: F,FT,FP FOUND FROM EQUATIONS INSERTED BELOW.
C          ISURF=-F FROM EQUATION;FT,FP NUMERICALLY DETERMINED.
C          ISURF=0: F IS FUNCTION OF PHI ONLY AND IS IN TABULAR FORM.
C                   FT,FP NUMERICALLY DETERMINED.
C          ISURF>0: F IS FUNCTION OF THETA AND PHI AND IS IN TABULAR
C                   FORM. FT,FP NUMERICALLY DETERMINED.
C
      IF(ISURF)20,40,30
20    CONTINUE
C
C        CALCULATE F,FT,FP FROM EQUATION TO BE INSERTED HERE
C
      DO 22 N=1,NMAX
      DO 22 M=1,MMAX
       DEN=COT(M)
       F(M,N)=0.5     /DEN
      IF(ISURF .EQ. -2)GO TO 22
        FT(M,N)=F(M,N)/COT(M)*SIT(M)
        FP(M,N)=0.0
22     CONTINUE
      IF(ISURF .EQ. -2)GO TO 100
      GO TO 110
40    CONTINUE
C
C       CASE OF ZERO PHI VARIATION
C
C
C
      READ(5,1002)(F(M,1),M=1,MMAX)
```

```
      MSTOP=0
      DO 49 M=1,MMAX
       IF(MSTOP .EQ. 1)GO TO 45
C
C        IF RHO=999.99999,EDGE HAS JUST BEEN PASED;VALUES BEYOND EDGE
C        WERE ASSIGNED ARBITRARILY.
C
       IF(F(M,1) .NE. 999.99999)GO TO 49
C
C        EXTRAPOLATE FOR RHO VALUES PAST REFLECTOR EDGE
C
        MSTOP=1
45      IF(M .GE. 4)F(M,1)=2.5*F(M-1,1)-2.*F(M-2,1)+.5*F(M-3,1)
        IF(M .EQ. 3)F(M,1)=2.*F(M-1,1)-F(M-2,1)
        IF(M .EQ. 2)F(M,1)=F(M-1,1)
49     CONTINUE
1002  FORMAT(7(1X,F9.5))
C
C        CALCULATE FT USING FORWARD & BACKWARD DIFFERENCES
C
C
      CALL DIFF(I,-1,  MMAX,1,F,      FT,DT,DP)
      DO 43 M=1,MMAX
C
C        DRHO/DPHI IDENTICALLY ZERO
C
43      FP(M,1)=0.
C
C        ASSIGNMENTS FOR N=2 TO NMAX
C
      DO 44 N=2,NMAX
      DO 44 M=1,MMAX
        F(M,N)=F(M,1)
        FT(M,N)=FT(M,1)
44      FP(M,N)=FP(M,1)
C
          GO TO 110
30     CONTINUE
C
C        VARIATION IN THETA AND PHI
C
100     CONTINUE
      READ(5,1002)((F(M,N),M=1,MMAX),N=1,NMAX)
      DO 31 N=1,NMAX
       MSTOP=C
      DO 32 M=1,MMAX
       IF(MSTOP .EQ. 1)GO TO 35
C
C        CHECK FOR RHO=999.99999
C
       IF(F(M,N) .NE. 999.99999)GO TO 32
C
C        EXTRAPOLATE RHO VALUES PAST REFLECTOR EDGE TO END OF INTEG GRID
C
        MSTOP=1
35      IF(M .GE. 4)F(M,N)=2.5*F(M-1,N)-2.*F(M-2,N)+.5*F(M-3,N)
        IF(M .EQ. 3)F(M,N)=2.*F(M-1,N)-F(M-2,N)
        IF(M .EQ. 2)F(M,N)=F(M-1,N)
32     CONTINUE
31    CONTINUE
C
C        USE BACK & FORWARD DIFFERENCES TO COMPUTE FT & FP
C
      CALL DIFF(I,-1,  MMAX,NMAX,F,      FT,DT,DP)
C
      CALL DIFF(I,1,  MMAX,NMAX,F,      FP,DT,DP)
C
110   CONTINUE
C
C        WRITE SOME SURFACE PARAMETERS
C
      DO 120 N=1,NMAX,15
      DO 120 M=1,MMAX
120   WRITE(6,2003)  M,N,F(M,N),FT(M,N),FP(M,N)
2003  FORMAT(1H ,2I3,3(1X,F10.4))
         RETURN
         END
```

```
CCCCCCCCCCCC                                                                  01030000
CCCCCCCCCCCC                                                                  01040000
C                                                                             01050000
C                                                                             01060000
      SUBROUTINE EDGEE3(NMAX,TEDGE)                                           01070000
C                                                                             01080000
C        CALCULATE THETA AT EDGE OF REFLECTOR USING EQUATION IN PHI           01090000
C        CAN HAVE DIFFERENT EQUATIONS FOR SECTIONS IN PHI OF GRID            01091000
C                                                                             01092000
      COMMON/GRID1/SIT(36),COT(36),SIP(91),COP(91),T(36),P(91)               01100000
      DIMENSION TEDGE(91)                                                     01110000
C                                                                             01120000
       DTR=0.01745329                                                         01125000
C        INSERT EQUATIONS HERE                                                01130000
C       NOTE: TEDGE MUST BE RETURNED IN DEGREES!!                            01131000
C                                                                             01140000
      DO 111 N=1,NMAX                                                         01150000
       IF((P(N)/DTR .GE. 315.)    .OR.  (P(N)/DTR .LE. 45.))GO TO 10         01150100
       IF((P(N)/DTR .GE. 45.) .AND. (P(N)/DTR .LE. 135.))GO TO 20            01150200
       IF((P(N)/DTR .GE. 135.) .AND. (P(N)/DTR .LE. 225.))GO TO 30           01150300
       IF((P(N)/DTR .GE. 225.) .AND. (P(N)/DTR .LE. 315.))GO TO 40           01150400
10      TEDGE(N)=ATAN(0.069927/ABS(COP(N)))                                   01150500
        GO TO 111                                                             01150600
20      TEDGE(N)=ATAN(0.069927/ABS(SIP(N)))                                   01150700
        GO TO 111                                                             01150800
30      TEDGE(N)=ATAN(0.069927/ABS(COP(N)))                                   01150900
        GO TO 111                                                             01151000
40      TEDGE(N)=ATAN(0.069927/ABS(SIP(N)))                                   01151100
111    TEDGE(N)=TEDGE(N)/DTR                                                  01170000
       RETURN                                                                 01180000
       END                                                                    01190000
```

```
      SUBROUTINE FINT(X,Y,F,R,MMAX,NMAX,STOT,I,J,KK,PC,IEDGE,TEDGE)          00010000
C                                                                             00020000
C        THIS SUB. NUMERICALLY INTEGRATES THE RADIATION INTEGRAL             00030000
C                                                                             00040000
      DIMENSION  X(36),Y( 91), R(36,91),TEDGE( 91),           RTEMP(2)       00050000
      COMPLEX F(36,91,3),STOT(3),T1,T2,T3,A,B,C,F12,F23,F14,SUM,TOT         00060000
      COMPLEX FVTEMP(2,3),AT,DELTA                                            00070000
C                                                                             00070100
C         SOME VARIABLE DEFINITIONS:                                          00070200
C                                                                             00070300
C      X,Y = THETA,PHI                                                        00070400
C      R = PATHLENGTH TERM                                                    00070500
C      F = VECTOR RELATED TO SURFACE CURRENTS                                 00070600
C      STOT = 3 COMPONENTS OF EVALUATED INTEGRAL (RETURNED TO MAIN)          00070700
C      TEDGE = THETA VALUES THAT SPECIFY REFLECTOR EDGE AS A FUNCTION        00070900
C              OF PHI                                                         00071000
C      IEDGE = INTEGER PARAMETER THAT SIGNIFIES PRESENCE OF EDGE AND         00071100
C              SPECIFIES HOW TEDGE IS OBTAINED.                               00071200
C      RTEMP,FVTEMP = TEMPORARY STORAGE FOR R,F                               00071300
C      NCE = PARAMETER THAT INDICATES FOR EACH PHI,WHEN AN EDGE THETA        00071400
C          HAS BEEN REACHED.                                                  00071500
C      TEAVG = AVERAGE OF EDGE THETA VALUES CORRESPONDING TO PHI AND         00071600
C            PHI+DELTA(PHI). THIS COMPROMISE RESULTS IN A SLIGHTLY           00071700
C            DISCONTINUOUS REFLECTOR EDGE.                                    00071800
C                                                                             00071900
      NXP1=NMAX+1                                                             00075000
      MXP1=MMAX+1                                                             00076000
C                                                                             00077000
C        INITIALIZE STOT                                                      00090000
C                                                                             00100000
      DO 10 L=1,3                                                             00220000
10    STOT(L)=(0.0,0.0)                                                       00230000
C                                                                             00231000
C        ENTER  PHI LOOP                                                      00232000
C                                                                             00233000
      DO 200 N=1,NMAX                                                         00240000
      NCE=0                                                                   00245000
11    DY=0.5*(Y(N+1)-Y(N))                                                    00250000
C                                                                             00251000
C        ENTER THETA LOOP (WITHIN PHI LOOP)                                   00252000
C                                                                             00253000
      DO 201 M=1,MMAX                                                         00260000
C                                                                             00264000
C        CHECK IF EDGE TO BE ENCOUNTERED ON THIS INTEG GRID                   00265000
C                                                                             00266000
      IF(IEDGE .EQ. 1)GO TO 79                                                00270000
C                                                                             00271000
C        IF EDGE PRESENT,COMPARE EACH THETA (X(M)) WITH THE EDGE VALUE       00272000
C        OF THETA FOR THAT PHI. AT FIRST HINT OF EDGE.INTEGRATION            00273000
C        PROCEDURE IS ALTERED.                                                00274000
C                                                                             00275000
C        X(1) MUST NOT BE > TEDGE(N) FOR ANY N                                00280000
      IF((TEDGE(N) .GE. X(M+1)) .AND. (TEDGE(N+1) .GE. X(M+1)))              00290000
     &GO TO 79                                                                00300000
       IF((X(M) .LE. TEDGE(N)) .AND. (TEDGE(N) .LT. X(M+1)))                 00310000
     &GO TO 30                                                                00320000
       IF((X(M) .LE. TEDGE(N+1)) .AND.  (TEDGE(N+1) .LT.                     00330000
     & X(M+1)))GO TO 30                                                       00340000
```

```
      WRITE(6,2003)I,M,N,TEDGE(N),X(M)
2003  FORMAT(1H ,'ERROR!!!-EDGE COORDINATES CHANGING TOO QUICKLY FOR',
     &'M=',I5,' & N=',I5,'.THETA AT EDGE IS ',F8.2,' RAD. AND T(M) IS',
     &1X,F8.2)
 30   CONTINUE
C
C        REPLACE X(M+1) WITH AVERAGE OF TEDGE(N) & TEDGE(N+1)
C
      TEAVG=(TEDGE(N)+TEDGE(N+1))/2.
C
C        IF TEAVG > X(M+1),CONTINUE TO INTEGRATE OVER THIS SLIGHTLY
C        LARGER DELTA S (PHASE ERROR STILL SMALL)
C
      TDIFF=X(M+1)-TEAVG
      IF(TDIFF .LE. ((X(M)-X(M+1))/4.))WRITE(6,2004)M,N
2004  FORMAT(1H ,'ERROR!!!!!!!!-TOO MUCH PHASE WIND AT EDGE FOR M=',
     &I5,' AND N=',I5,' INTEGRATION CONTINUING')
C
C        RECOMPUTE DX
C
      DS=DY*(TEAVG-X(M))
      NCE=1
C     IF(J .NE. 5)GO TO 40
C      TEAVGD=TEAVG/0.01745329
C      WRITE(6,2002)TEAVGD,M,N
C002   FORMAT(1H ,'THETA EDGE AVG IS ',F10.4,' FOR M,N = ',2I5)
C0    CONTINUE
      NP1=N+1
      NNN=N
      U=  (TEAVG-X(M))/(X(M+1)-X(M))-1
C
C        FOR UPPER CORNERS OF (TRAPEZOIDAL) DELTA(S),INTERPOLATE R
C        AND F TO TEAVG. USE 3 TERM INTERPOLATION ;2 TERMS AT LEAST.
C
      DO 51 NN=NNN,NP1
C
C        STORE PATHLENGTH VALUE OF ORIGINAL DELTA(S) CORNERS.
C
      RTEMP(NN-NNN+1)=R(M+1,NN)
C
C        USE 2 TERMS IN INTERPOLATION IF M<3
C
       IF(M .LT. 3)GO TO 90
      R(M+1,NN)=R(M+1,NN)+U*(R(M+1,NN)-R(M,NN))+.5*U*(U-1)*(R(M+1,NN)
     &-2.*R(M   ,NN)+R(M-1,NN))
      GO TO 91
 90   R(M+1,NN)=R(M+1,NN)+U*(R(M+1,NN)-R(M,NN))
 91    DO 54 L=1,3
C
C        STORE F VECTOR VALUE OF ORIGINAL DELTA(S) CORNERS.
C
       FVTEMP(NN-NNN+1,L)=F(M+1,NN,L)
C
C        USE 2 TERMS IN INTERPOLATION IF M<3
C
       IF(M .LT. 3)GO TO 92
        F(M+1,NN,L)=F(M+1,NN,L)+U*(F(M+1,NN,L)-F(M,NN,L))+.5*U*(U-1)*
     &(F(M+1,NN,L)-2.*F(M,NN,L)+F(M-1,NN,L))
      GO TO 54
 92   F(M+1,NN,L)=F(M+1,NN,L)+U*(F(M+1,NN,L)-F(M,NN,L))
 54   CONTINUE
C     IF(J .NE. 5)GO TO 56
C       WRITE(6,2007)M,NN,R(M-1,NN),R(M,NN),RTEMP(NN-NNN+1),R(M+1,NN)
C    &FVTEMP(NN-NNN+1,2),F(M+1,NN,2)
C007  FORMAT(1H ,2I3,5(1X,E12.6))
C6     CONTINUE
 51   CONTINUE
      GO TO 80
 79   DS=DY*(X(M+1)-X(M))
 80   CONTINUE
C
C        A LEAST SQUARES BEST FIT IS APPLIED TO R ONTO DELTA(S)
C
      R1=R(M+1,N+1)-R(M,N)
      R2=R(M,N+1)-R(M+1,N)
      R3=R(M,N)+R(M+1,N)
      BE=0.5*(R1+R2)
      CE=0.5*(R1-R2)
      AL=0.5*(R3-CE)
C
C        EVALUATE DELTA(I) EXPRESSION TO FIND SCATTERED FIELD
C        CONTRIBUTION FROM EACH DELTA(S).
C
      IF(ABS(BE)-0.01)100,100,110
100   F3I=BE*0.33333333
      F1I=BE*0.5
      F1R=1.0-F1I*F3I
      F3R=0.5-BE*BE/8.0
      GO TO 140
110   SINBE=SIN(BE)
      COSBE=COS(BE)
      F1R=SINBE/BE
      F1I=(1.0-COSBE)/BE
      F3R=F1R-F1I/BE
      F3I=(F1R-COSBE)/BE
140   IF(ABS(CE)-0.01)150,150,160
150   F4I=0.33333333*CE
      F2I=0.5*CE
      F2R=1.0-F2I*F4I
      F4R=0.5-CE*CE/8.0
      GO TO 170
160   SINCE=SIN(CE)
      COSCE=COS(CE)
      F2R=SINCE/CE
      F2I=(1.0-COSCE)/CE
      F4R=F2R-F2I/CE
      F4I=(F2R-COSCE)/CE
170   SINAL=SIN(AL)
      COSAL=COS(AL)
      F12=CMPLX(F1R*F2R-F1I*F2I,F1I*F2R+F2I*F1R)
      F23=CMPLX(F2R*F3R-F2I*F3I,F2I*F3R+F3I*F2R)
      F14=CMPLX(F1R*F4R-F1I*F4I,F1I*F4R+F4I*F1R)
      DO 202 L=1,3
C
C        A LEAST SQUARES BEST FIT IS APPLIED TO F ONTO DELTA(S)
C
```

```
      T1=F(M+1,N+1,L)-F(M,N,L)
      T2=F(M,N+1,L)-F(M+1,N,L)
      T3=F(M,N,L)+F(M+1,N,L)
      B=T1+T2
      C=T1-T2
      A=T3-0.5*C
      SUM=A*F12+B*F23+C*F14
      TOT=CMPLX(COSAL,SINAL)*SUM*DS
C
C       ADD PATTERN CONTRIBUTION FROM EACH DELTA(S) OF INTEG GRID.
C
202   STOT(L)=STOT(L)+TOT
      IF(NCE .EQ. 0)GO TO 201
C
C       FOR NCE=1,RE-ASSIGN STORED R & F VALUES AND EXIT THETA LOOP
C
       DO 191 NN=NNN,NP1
        R(M+1,NN)=RTEMP(NN-NNN+1)
       DO 191 L=1,3
191     F(M+1,NN,L)=FVTEMP(NN-NNN+1,L)
       GO TO 200
201    CONTINUE
200   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SETUP(NG1,I,JMAX,KMAX,MMAX,NMAX)
C
C        GIVEN THE # ANGLE VALUES,AN INITIAL VALUE AND AN INCREMENT,
C        THIS SUB. ESTABLISHES THE OUTPUT GRID(#2) AND UP TO 21
C        INTEGRATION GRIDS. IT ALSO PRCOMPUTES TRIG VALUES ON ALL
C        POINTS OF THE GRIDS.
C
      DIMENSION MM(21),NN(21),T(36,21),P( 91,21)
      COMMON/GRID1/SIT(36),COT(36),SIP( 91),COP( 91),TG1(36),PG1( 91)
      COMMON/GRID2/SITT(181),COTT(181),SIPP(5),COPP(5),TT(181),PP(5)
C
C
C        SOME VARIABLE DEFINITIONS:
C
C        T,P = THETA,PHI OF INTEG GRID
C        MMAX,NMAX = # OF T,P VALUES
C        TT,PP = BIGTHETA,BIGPHI OF OUTPUT GRID
C        JMAX,KMAX = # OF TT,PP VALUES
C        NG1 = # INTEGRATION GRIDS
C
C
C       READ IN OUTPUT GRID
C
      READ(5,1001)JMAX,TT1,DTT
      IF(DTT)10,10,10
10    DO 15 J=1,JMAX
      FJ=J-1
15    TT(J)=TT1+DTT*FJ
      GO TO 30
20    READ(5,1002)(TT(J),J=1,JMAX)
30    READ(5,1001)KMAX,PP1,DPP
      IF(DPP)40,40,40
40    DO 45 K=1,KMAX
      FK=K-1
45    PP(K)=PP1+FK*DPP
      GO TO 60
50    READ(5,1002)(PP(K),K=1,KMAX)
60    CONTINUE
C
C        INPUT INTEGRATION GRIDS
C
      READ(5,1001)NG1
      DO 100 I=1,NG1
      READ(5,1001)MM(I),T1,DT
      MMM=MM(I)
      IF(DT)110,110,110
110   DO 115 M=1,MMM
      FM=M-1
115   T(M,I)=T1+FM*DT
      GO TO 130
120   READ(5,1002)(T(M,I),M=1,MMM)
130   READ(5,1001)NN(I),P1,DP
      NNN=NN(I)
      IF(DP)140,140,140
140   DO 145 N=1,NNN
      FN=N-1
145   P(N,I)=P1+FN*DP
      GO TO 100
```

```
150   READ(5,1002)(P(N,I),N=1,NNN)
100   CONTINUE
C
C       PRINT OUT GRID DATA
      WRITE(6,2001)
      WRITE(6,2006)(J,TT(J),J=1,JMAX)
      WRITE(6,2002)
      WRITE(6,2006)(K,PP(K),K=1,KMAX)
      WRITE(6,2004)NG1
      DO 155 I=1,NG1
      WRITE(6,2005)I
      MMM=MM(I)
      NNN=NN(I)
      WRITE(6,2006)(M,T(M,I),M=1,MMM)
      WRITE(6,2002)
155   WRITE(6,2006)(N,P(N,I),N=1,NNN)
C
C       COMPUTE GRID2 TABLE
C
      DTR=0.017453293
      DO 200 J=1,JMAX
      TT(J)=TT(J)*DTR
      SITT(J)=SIN(TT(J))
      COTT(J)=COS(TT(J))
200   CONTINUE
      DO 210 K=1,KMAX
      PP(K)=PP(K)*DTR
      COPP(K)=COS(PP(K))
      SIPP(K)=SIN(PP(K))
210   CONTINUE
C
C       COMPUTE GRID1 TABLES
C
      I=0
C
C       FOR NG1 >1 RE-ENTER SUB. HERE
C
      ENTRY RESET(I,MMAX,NMAX)
      I=I+1
      MMAX=MM(I)
      DO 400 M=1,MMAX
      TG1(M)=DTR*T(M,I)
      SIT(M)=SIN(TG1(M))
      COT(M)=COS(TG1(M))
400   CONTINUE
      NMAX=NN(I)
      DO 410 N=1,NMAX
      PG1(N)=DTR*P(N,I)
      SIP(N)=SIN(PG1(N))
      COP(N)=COS(PG1(N))
410   CONTINUE
      RETURN
C
1001  FORMAT(I5,2F10.2)
1002  FORMAT(8F10.2)
2001  FORMAT(1H ,'  THE FOLLOWING OUTPUT GRID HAS BEEN ESTABLISHED',/,
     &'        ALL ANGLES IN DEGREES',//)
2002  FORMAT(//)
2004  FORMAT('        THE FOLLOWING',I3,' INTEGRATION GRIDS HAVE BEEN ',
     &'ESTABLISHED',/)
2005  FORMAT(1H0,'  SEGMENT NUMBER',I3,/)
2006  FORMAT(    5(I4,    F8.3))
      END
```

```
      SUBROUTINE FIELDS
C
C        THIS SUB. FINDS THE MAGNETIC FIELD ON A SURFACE S BY
C        EVALUATING A SPHERICAL WAVE EXPANSION.THE WAVE COEFICIENTS
C        AS CALCULATED BY ANOTHER PROGRAM ARE OBTAINED VIA A DISK
C        DATA SET
C
      COMMON/GRID1/SIT(36),COT(36),SIP( 91),COP( 91),T(36),P( 91)
      COMMON/GRID2/SITT(181),COTT(181),SIPP(5),COPP(5),TT(181),PP(5)
      DIMENSION F( 81),G( 81)
      DIMENSION NAME(18)
      DIMENSION A( 80,2),B( 80,2)
C
C         SOME VARIABLE DEFINITIONS:
C
C         LMAX = MAXIMUM MODE ORDER
C         F,G = VECTORS OF LEGENDRE POLYNOMIAL WEIGHTS EVALUATED FOR EACH
C               MODE COMPONENT. THE ARGUEMENT OF THE POLY IS THE COS OF
C               A POLAR ANGLE.
C         A,B = TE & TM WAVE COEFFICIENTS
C
C
C        INPUT WAVE COEFFICIENTS FROM COMPUTER DISK
C
      READ(9,1001)NAME
      WRITE(6,2001)NAME
      READ(9,1002)LMAX,MCOMP
      READ(9,1001)NAME
      READ(9,1003)(J,A(J,1),A(J,2),B(J,1),B(J,2),J=1,LMAX)
      FMC=MCOMP
      WRITE(6,2003)MCOMP
      WRITE(6,2002)NAME
      WRITE(6,2004)(J,A(J,1),A(J,2),B(J,1),B(J,2),J=1,LMAX)
      RETURN
CCCCC
CCCCC
C        ENTRY POINT FOR MAGNETIC FIELDS AT FINITE R
C
      ENTRY FIELD1(MMAX,NMAX,HR,HT,HP,R)
      DIMENSION R(36,91)
      COMPLEX HR(36,91),HT(36,91),HP(36,91)
C
C         MORE VARIABLES:
C
C         R = RHO
C         HR,HT,HP = INCIDENT MAGNETIC FIELD COMPONENTS (RETURNED
C         TO FIELD1)
C         MMAX,NMAX = # OF THETA,PHI POINTS ON INTEG GRID
C
      IENT=1
      M=0
C
C         ENTER THETA LOOP
C
    1 M=M+1
      SN=SIT(M)
      Z=COT(M)
      TOUT=T(M)
      GO TO 99
C
C        ENTRY POINT FOR ELECTRIC FIELDS AT INFINITE R
C
      ENTRY FIELD2(JO,KO,ETTO,EPPO)
      COMPLEX ETTO,EPPO
C
C          STILL MORE VARIABLES:
C
C          JO,KO = POINT ON OUTPUT GRID
C          ETTO,EPPO = FIELD VALUES AT JO,KO (RETURNED TO MAIN)
C
      IENT=2
      SN=SITT(JO)
      Z=COTT(JO)
      TOUT=TT(JO)
C
C        FORM LEGENDRE FUNCTION TABLES FOR M-TH THETA VALUE
C
   99 IF(ABS(SN)-.00001)200,100,100
  100 DO 105 N=1,MCOMP
  105 F(N)=0
C
C          OBTAIN F & G VECTORS FOR A GIVEN ORDER OF AZIMUTHAL VARIATION
C
      NC=LMAX+1
      CALL LEGEND(NC,MCOMP,Z,F)
      DO 110 N=1,LMAX
      T1=N-MCOMP+1
      T2=N+1
      G(N)=T1*F(N+1)-T2*Z*F(N)
  110 G(N)=G(N)/SN
      DO 115 N=1,LMAX
  115 F(N)=F(N)/SN
C
C          FOR NEAR FIELD CASE,MUST CONSIDER RADIAL DEPENDENCE OF FIELDS
C          AS DEFINED BY THE SPHERICAL HANKEL FUNCTION.
C
      GO TO (300,500),IENT
C
C        SPECIAL EQUATIONS FOR THETA=0,180 DEG
C
  200 IF(MCOMP-1)210,220,210
  210 DO 215 N=1,LMAX
      F(N)=0
  215 G(N)=0
      GO TO (300,500),IENT
  220 DO 225 N=1,LMAX
      FN=N*(N+1)
      F(N)=FN/2.0
  225 G(N)=FN/2.0
      IF(TOUT-1.57)250,250,230
  230 DO 235 N=1,LMAX,2
      F(N+1)=-F(N+1)
  235 G(N)=-G(N)
  250 GO TO (300,500),IENT
C
C        FOR EACH PHI THE HANKEL FUNCTIONS ARE EVALUATED AT THE
```

```
C       CORRESPONDING VALUE OF R FOR EACH MODE COMPONENT.
C
300   DO 450 N=1,NMAX
      HRR=0
      HRI=0
      HTR=0
      HTI=0
      HPR=0
      HPI=0
      CALL SPHANK(1,R(M,N),SOR,SOI)
      DO 400 L=1,LMAX
      FL=L
      FN=(FL+1.0)/R(M,N)
      NC =L+1
      CALL SPHANK(NC,R(M,N),SIR,SII)
C
C       TIME FOR SOME GLOP
C        DETERMINE THE FIELD AT EACH THETA,PHI ON THE INTEG GRID BY
C        FIRST OBTAINING THE PRODUCT OF COEFFICIENTS AND THE SPHANK
C        VALUES FOR EACH MODE COMPONENT.
C
      F1R=-A(L,1)*SIR+A(L,2)*SII
      F1I=-A(L,1)*SII-A(L,2)*SIR
      F2R=FN*(A(L,1)*SOI+A(L,2)*SOR)
      F2I=FN*(A(L,2)*SOI-A(L,1)*SOR)
      F3R=-B(L,1)*SOR+B(L,2)*SOI
      F3I=-B(L,1)*SOI-B(L,2)*SOR
C
C         THEN MULTIPLY BY THE CORRESPONDING MODE LEGENDRE POLY VALUES.
C         FOR EACH THETA,PHI THE ABOVE VALUES ARE SUMMED OVER ALL MODE
C         COMPONENTS.
C
      HRR=HRR+F2R*F(L)*FL
      HRI=HRI+F2I*F(L)*FL
      F(L)=F(L)*FMC
      HTR=HTR+G(L)*F1R+F2R*G(L)+F3R*F(L)
      HTI=HTI+F1I*G(L)+F2I*G(L)+F3I*F(L)
      HPR=HPR-F1R*F(L)-F2R*F(L)-F3R*G(L)
      HPI=HPI-F1I*F(L)-F2I*F(L)-F3I*G(L)
      SOR=SIR
      SOI=SII
400   CONTINUE
C
C        FINALLY,THE PHI EXPANSION TERM IS INTRODUCED.COMPLETING THE
C        EXPANSION.
C
      HRR=HRR*SN*COS(FMC*P(N))
      HRI=HRI*SN*COS(FMC*P(N))
      HR(M,N)=CMPLX(HRR,HRI)
      HTR=HTR*COS(FMC*P(N))
      HTI=HTI*COS(FMC*P(N))
      HT(M,N)=CMPLX(HTR,HTI)
      HPR=HPR*SIN(FMC*P(N))
      HPI=HPI*SIN(FMC*P(N))
      HP(M,N)=CMPLX(HPR,HPI)
C      WRITE(6,2005)M,N,HR(M,N),HT(M,N),HP(M,N)
2005   FORMAT(1H ,'H-FIELDS',2I5,6F12.5)
450   CONTINUE
      IF(M-MMAX)1,460,460
460   RETURN
C
C        FAR FIELD CASE
C
500   CONTINUE
      ETR=0.0
      ETI=0
      EPR=0
      EPI=0
      DO 550 L=1,LMAX
      F(L)=F(L)*FMC
C
C        SUM PRODUCT OF COEFICIENTS AND LEGENDRE WEIGHTS OVER ALL MODE
C        COMPONENTS. THIS IS DONE FOR EACH OUTPUT GRID POINT.
C
      ETR=ETR+A(L,1)*F(L)+B(L,1)*G(L)
      ETI=ETI+A(L,2)*F(L)+B(L,2)*G(L)
      EPR=EPR+A(L,1)*G(L)+B(L,1)*F(L)
      EPI=EPI+A(L,2)*G(L)+B(L,2)*F(L)
550   CONTINUE
C
C        INTRODUCE BIGPHI VARIATION TO COMPLETE EXPANSION.
C
      ETR=ETR*SIN(FMC*PP(KO))
      ETI=ETI*SIN(FMC*PP(KO))
      ETTO=CMPLX(ETR,ETI)
      EPR=EPR*COS(FMC*PP(KO))
      EPI=EPI*COS(FMC*PP(KO))
      EPPO=CMPLX(EPR,EPI)
      RETURN
C
1001  FORMAT(18A4)
1002  FORMAT(5I5)
1003  FORMAT(I5,2E17.8,2X,2E17.8)
2001  FORMAT(1H0,'      FIELD DATA INPUT IN THE FORM OF SPHERICAL WAVE ',
     &'COEFFICIENTS',/,/5X,18A4,/)
2002  FORMAT(1H05X,18A4,///,20X,4HA(N),32X,4HB(N),/,
     &5H     N,7X4HREAL,13X,4HIMAG,15X4HREAL,13X,4HIMAG)
2003  FORMAT(1H ,' AZIMUTHAL ORDER MCOMP=',I2)
2004  FORMAT(I5,2E17.8,2X,2E17.8)
      END
```

```
      SUBROUTINE PATHL(RHO,J,K,MMAX,NMAX,GAM)
C
C        THIS SUB. COMPUTES PATH LENGTH FUNCTION GAMMA
C        FOR EACH POINT ON THE INTEC GRID AND IN THE DIRECTION OF AN
C        OUTPUT GRID POINT.
C        I.E.,THE PATHLENGTH IS A FUNCTION OF 4 VARIABLES.
C
      DIMENSION RHO(36,91),GAM(36,91)
      COMMON/GRID1/SIT(36),COT(36),SIP( 91),COP( 91),T(36),P( 91)
      COMMON/GRID2/SITT(181),COTT(181),SIPP(5),COPP(5),TT(181),PP(5)
C
C        VARIABLES:
C
C        J,K = OUTPUT GRID POINT
C        GAM = PATHLENGTH TERM (RETURNED TO MAIN)
C
      DO 10 M=1,MMAX
      T1=SIT(M)*SITT(J)*COPP(K)
      T2=SIT(M)*SITT(J)*SIPP(K)
      T3=COT(M)*COTT(J)-1.0
      DO 10 N=1,NMAX
      GAM(M,N)=RHO(M,N)*(T1*COP(N)+T2*SIP(N)+T3)
10    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VECTOR(X,Y,AMP,PHI)
C
C        THIS SUB. CONVERTS COMPLEX VALUES TO POLAR FORM
C        AMP & PHI RETURNED
C
      C=0.0
      IF(X)100,200,300
100   IF(Y)110,120,120
110   C=360.0
120   PHI=ATAN(Y/X)*57.29577951+180.0-C
      GO TO 400
200   IF(Y)210,220,230
210   PHI=-90.0
      GO TO 400
220   PHI=0.0
      GO TO 400
230   PHI=90.0
      GO TO 400
300   PHI=ATAN(Y/X)*57.29577951
400   AMP=SQRT(X*X+Y*Y)
      RETURN
      END
```

```
      SUBROUTINE SPHANK(N,R,X,Y)
C
C        THIS SUB. COMPUTES VALUE OF SPHERICAL HANKEL FUNCTION TIMES
C        THE FACTOR (-J)**(N+1)*RHO*EXP(J*RHO). THIS FACTOR IS THE
C        RECIPROCAL OF THE FAR FIELD SPHERICAL HANKEL FUNCTION. THE
C        FAR FIELD DEPENDENCE IS BUILT-IN TO THE SPHERICAL WAVE
C        COEFFICIENTS AND MUST BE REMOVED FOR NEAR FIELD CASES.
C        THE HANKEL FUNCTION PROVIDES THE NEAR FIELD RADIAL VARIATION
C        OF THE PATTERN.
C
C
      DOUBLE PRECISION TERM,AI,AR
C
C        VARIABLES:
C
C        N = MODE ORDER + 1
C        R = ARGUMENT OF FUNCTION = PHASE CONSTANT * RHO
C        X,Y = REAL & IMAG COMPUTED VALUES (RETURNED TO FIELD1)
C
      AR=0
      AI=0
      PI=3.1415927
      K=0
      TERM=1
      GO TO 100
20    K=K+1
      T1=N+K
      T2=N-K+1
      T3=2*K
      TERM=TERM*T1*T2/T3
      TERM=TERM/R
C
C        THE NUMERICAL TECHNIQUE CAN BLOW-UP FOR R<50 (ROUGHLY) AND IS
C        THEREFORE TRUNCATED BEFORE OVERFLOW OCCURS. THE IMAG COMPONENT
C        SHOULD BLOW-UP (A MATHEM. PROPERTY OF THE FUNCTION)  AND  IS
C        PROPERLY FOUND TO DO SO. THE REAL VALUE SHOULD GO TO ZERO AS
C        THE IMAG VALUE BECOMES VERY LARGE.HOWEVER, THIS IS NOT THE CASE
C        IT TOO BLOWS-UP.
C
      IF(DABS(TERM) .LT. .000001)GO TO 1000
      GO TO(200,100),IGO
100   AR =AR+TERM
      IGO=1
      IF(K-N)20,1000,1000
200   AI=AI-TERM
      IGO=2
      TERM=-TERM
      IF(K-N)20,1000,1000
1000  X=AR
      Y=AI
      RETURN
      END
```

```
      SUBROUTINE ADJUST(PHI)
C
C        THIS SUB. SHIFTS PHI UNTIL IT LIES IN THE RANGE -180,180
C
1     IF(PHI-180.)20,20,10
10    PHI=PHI-360
      GO TO 1
20    IF(PHI+180.)30,40,40
30    PHI=PHI+360
      GO TO 20
40    RETURN
      END
```

```
      SUBROUTINE LEGEND(NMAX,M,Z,VAL)
C
C        THIS SUB. CALCULATES VALUES OF THE ASSOCIATED LEGENDRE
C        FUNCTION WITH INDICES N=M TO N=NMAX.DO NOT USE NMAX LESS
C        THAN M+1.VALUES CHECKED WITH TABULATED VALUES TO 5 PLACES
C        THRU N=56 FOR M=1,AND N=10 FOR M=5.
C
      DIMENSION VAL( 81)
      DOUBLE PRECISION TERM1,TERM2,TERM3,ZD
C
C        VARIABLES:
C
C        NMAX = MODE ORDER + 1
C        M = MCOMP = ORDER OF AZIMUTHAL VARIATION
C        Z = ARGUMENT OF FUNCTION = COS(THETA) OR COS(BIGTHETA)
C        VAL = VALUES OF LEGENDRE POLY FOR EACH MODE (RETURNED TO
C               FIELD1,FIELD2)
C
      ZD=Z
      FM=M
      TERM1=0.0
      IF(M)8,8,9
8     TERM2=1.0
      GO TO 11
9     KMAX=2*M-1
      TERM2=(1.0-ZD*ZD)**(FM/2.0)
      DO 10 K=1,KMAX,2
      FK=K
10    TERM2=TERM2*(2.0*FM-FK)
      VAL(M)=TERM2
11    NN=NMAX-1
      DO 20 N=M,NN
      FN=N
      TERM3=((2.0*FN+1.0)*ZD*TERM2-(FN+FM)*TERM1)/(FN-FM+1.0)
      VAL(N+1)=TERM3
      TERM1=TERM2
      TERM2=TERM3
20    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DIFF(II,J,  MMAX,NMAX,F,      FX,DT,DP)
C
C        THIS SUB. RETURNS TO SURF THE VALUE OF FT OR FP. THESE DERIVS
C        ARE COMPUTED USING FORWARD AND BACKWARD DIFFERENCES.
C
      DIMENSION F(36,91),FX(36,91),              ITERM(21,20)
C
C        SOME VARIABLES USED:
C
C        ITERM = COEFFICIENTS WITHIN A TERM OF THE DIFFERENCE EQUATION
C        NT = # TERMS IN EQUATION
C        II = INTEG GRID #
C        J = INDICATOR OF WHETHER FT OR FP IS BEING SOUGHT
C        FX = EITHER FT OR FP
C
      DTR=0.01745329
C
C        FOR II=1 CALCULATE TERM COEFFICIENT FOR NUMERICAL DERIVATIVE
C        FORMULA. THESE COEFFICIENTS ARE IDENTICAL TO THOSE OF A
C        BINOMIAL EXPANSION.
C
      IF(II .NE. 1)GO TO 10
       ITERM(1,1)=1
       ITERM(2,1)=-1
       DO 11 JJ=2,20
       JJP1=JJ+1
       DO 12 I=1,JJP1
        IF(I .NE. 1)GO TO 13
         ITERM(1,JJ)=1
         GO TO 12
13      IF(I .NE. JJP1)GO TO 14
         ITERM(I,JJ)=               -ITERM(I-1,JJ-1)
         GO TO 12
14      ITERM(I,JJ)=ITERM(I,JJ-1)-ITERM(I-1,JJ-1)
12      CONTINUE
11      CONTINUE
C     DO 71 JJ=1,10
C      JJP1=JJ+1
C      WRITE(6,2001)(ITERM(I,JJ),I=1,JJP1)
2001   FORMAT(1H ,21I5)
C1    CONTINUE
10      CONTINUE
C
C        IF J=-1,COMPUTE FP;OTHERWISE COMPUTE FT
C
      IF(J .EQ. -1)GO TO 60
50    CONTINUE
C
C        CALCULATE FP
C
       DEN=DP*DTR*0.0254
C
C        ENTER THETA & PHI LOOPS (PHI WITHIN THETA). PERFORM CALCULATION
C        FOR EACH POINT ON INTEG GRID.
C
      DO 54 M=1,MMAX
      DO 54 N=1,NMAX
       FX(M,N)=0.
```

```
C
C       CALCULATE # TERMS USED IN DIFFERENCE EQN. THE ENTIRELY
C       EMPIRICAL TECHNIQUE OF CONVERTING DELTA(RHO) TO INCHES AND
C       DIVIDING BY 100 IS USED TO DETERMINE NT.
C
      IF(N .LT. NMAX)NT=2+ABS(F(M,N+1)-F(M,N))/DEN/100
      IF(N .EQ. NMAX)NT=2+ABS(F(M,N)-F(M,N-1))/DEN/100
      IF(NT .GT. 20)NT=20
C
C       USE BACK DIFF. IF POSSIBLE. IF NOT,USE METHOD THAT ALLOWS MOST
C       TERMS (COMPARE BACK DIFF. & FORWARD DIFF.)
C
      IF((N-NT) .GE. 1)GO TO 51
       NTB=N-1
      IF((N+NT) .LE. NMAX)GO TO 58
       NTF=NMAX-N
      IF(NTB .GE. NTF)GO TO 57
       NT=NTF
       GO TO 58
57     NT=NTB
       GO TO 51
C
C       DETERMINE FP USING FORWARD DIFF.
C
58      DO 52 NN=1,NT
         NNP1=NN+1
         SUM=0.
        DO 53 I=1,NNP1
53       SUM=SUM+ITERM(NNP1+1-I,NN)*F(M,N+I-1)
         FX(M,N)=FX(M,N)+SUM/FLOAT(NN)*(-1)**(NN+1)
52     CONTINUE
       FX(M,N)= FX(M,N)/DP/DTR
      GO TO 54
C
51    CONTINUE
C
C      DETERMINE FP USING BACK DIFF.
C
     DO 55 NN=1,NT
      NNP1=NN+1
       SUM=0.
     DO 56 I=1,NNP1
56    SUM=SUM+ITERM(I,NN)*F(M,N-I+1)
      FX(M,N)=FX(M,N)+SUM/FLOAT(NN)
55   CONTINUE
     FX(M,N)=FX(M,N)/DP/DTR
54   CONTINUE
       RETURN
60      CONTINUE
C
C       COMPUTE FT
C
      DEN=DT*DTR*0.0254
C
C       ENTER THETA & PHI LOOPS (THETA WITHIN PHI). PERFORM COMPUTATION
C       FOR EACH POINT ON INTEG GRID
C
     DO 64 N=1,NMAX
      DO 64 M=1,MMAX
       FX(M,N)=0.
C
C        CALCULATE # TERMS
C
      IF(M .LT. MMAX)NT=2+ABS(F(M+1,N)-F(M,N))/DEN/100
      IF(M .EQ. MMAX)NT=2+ABS(F(M,N)-F(M-1,N))/DEN/100
       IF(NT .GT. 20)NT=20
C
C        AGAIN,USE BACK DIFF IF POSSIBLE.
C
       IF((M-NT) .GE. 1)GO TO 61
        NTB=M-1
       IF((M+NT) .LE. MMAX)GO TO 68
        NTF=MMAX-M
       IF(NTB .GE. NTF)GO TO 67
        NT=NTF
        GO TO 68
67      NT=NTB
        GO TO 61
C
C        DETERMINE FT USING FORWARD DIFF.
C
68     DO 62 NN=1,NT
        NNP1=NN+1
        SUM=0.
        DO 63 I=1,NNP1
63       SUM=SUM+ITERM(NNP1+1-I,NN)*F(M+I-1,N)
        FX(M,N)=FX(M,N)+SUM/FLOAT(NN)*(-1)**(NN+1)
62      CONTINUE
       FX(M,N)=FX(M,N)/DT/DTR
C     IF(N .EQ. 1)WRITE(6,2002)M,N,F(M,N),FX(M,N),NT
2002  FORMAT(1H ,2I5,2(1X,F10.4),I5)
       GO TO 64
61     CONTINUE
C
C        DETERMINE FT USING BACK DIFF.
C
       DO 65 NN=1,NT
        NNP1=NN+1
        SUM=0.
       DO 66 I=1,NNP1
66      SUM=SUM+ITERM(I,NN)*F(M-I+1,N)
        FX(M,N)=FX(M,N)+SUM/FLOAT(NN)
65     CONTINUE
       FX(M,N)=FX(M,N)/DT/DTR
C     IF(N .EQ. 1)WRITE(6,2002)M,N,F(M,N),FX(M,N),NT
64     CONTINUE
         RETURN
         END
```

```
C        THIS IS A SPERICAL WAVE EXPANSION PROGRAM-ORTHOGONALITY
C        TECHNIQUE.THE SPHERICAL WAVE COEFFICIENTS FOR THE M-TH
C        FOURIER COMPONENT OF A RADIATION PATTERN ARE COMPUTED.
C        THE EXPANSION MATCHES THE FAR-FIELD INPUT PATTERN WITH
C        THE THE FAR-FIELD FORM OF SPHERICAL WAVES.
C
      DIMENSION NAME(18),NAMEJ(18)
      DIMENSION PA(80),PB(80)
      DIMENSION T(121),E(121),EP(121),H(121),HP(121)
      DIMENSION F(80,121),G(80,121),PM(81)
      DIMENSION A(121,2),B(121,2)
      DIMENSION ACOE(80,2),BCOE(80,2)
      DIMENSION AOUT(1,2),BOUT(1,2)
      DTR=0.017453293
C
C        SOME VARIABLE DEFINITIONS:
C
C     T,E,EP,H,HP=INPUT PATTERN PARAMETERS
C     JIN(<,=121)=# THETA INPUT VALUES
C     NMAX(<,=80) = MAXIMUM MODE ORDER
C     ACOE,BCOE = REAL & IMAG COMPONENTS OF TE AND TM WAVE COEFFICIENTS
C                 WRITTEN ON DISK
C     A,B = AMP & PHASE OF INPUT FIELDS VALUES * DELTA(THETA)
C     F,G =MATRICES (NMAX BY JIN OR NMAX BY JOUT OF VALUES RELATED TO
C         ASSOCIATED LEGENDRE POLYNOMIALS
C         G PROP. TO DERIVATIVE WRT THETA OF N-TH POLY.
C         F PROP. TO N-TH POLY.
C
C        INPUT M-TH COMPONENT OF INPUT PATTERN
C
      READ(5,3001)NAMEJ
2223   FORMAT(18A4)
      READ(5,1002)MCOMP,NMAX
2224   FORMAT(2I5)
C        CONVERT TO AMP AND PHASE
C
C
C        DETERMINE OUTPUT POWER THRU 10 DEG OF OUTPUT. ASSUME
C        DELTA(OUTPUT THETA) IS 1.0 DEG.
C
1001   FORMAT(3(1X,F9.5))
      READ(5,3001)NAME
      READ(5,1002)JIN,IC1,IC2,IC3,IC4
1002   FORMAT(   6I5)
C
C        IF IC3>0,CALCULATE INPUT PATTERN PARAMETERS FROM EQUATION
C
       IF(IC3)113,113,133
C
C        IF IC3<,=0 AND IC4>0,READ IN ONLY T,E,EP. I.E.,PATTERN IS CIRC.
C
113   IF(IC4)153,153,163
163    READ(5,1001)(T(M),E(M),EP(M),M=1,JIN)
      DO 173 M=1,JIN
       H(M)=E(M)
173    HP(M)=EP(M)
      GO TO 143
153   READ(5,1003)(T(M),E(M),EP(M),H(M),HP(M),M=1,JIN)
       GO TO 143
133    T(1)=0.0
       TR=0.0
       DELT=1.0
      DO 123 M=1,JIN
C
C        INSERT EQUATIONS FOR INPUT PATTERN HERE. NOTE ABOVE
C        INITIALIZATIONS.
C
         E(M)=SIN(5.0*TR)
         H(M)= E(M)
         TR=TR+DELT*DTR
         T(M+1)=T(M)+DELT
123   CONTINUE
1003   FORMAT(5(1X,F9.5))
143    CONTINUE
C
C       FOR IC2 > 0 NEGLECT PHASE
C
      IF(IC2)40,40,30
30    DO 35 J=1,JIN
      EP(J)=0.
35    HP(J)=0.
40    CONTINUE
C
C
C       FOR IC1 < OR = 0 CONVERT FROM DB TO VOLTS
C
      IF(IC1)10,10,20
10    DO 15 J=1,JIN
      E(J)=10.0**(E(J)/20.0)
15    H(J)=10.0**(H(J)/20.0)
20    CONTINUE
C
C       PRINT OUT INPUT PATTERN
C
      WRITE(6,2001)NAMEJ
      WRITE(6,2006)MCOMP
       WRITE(6,2001)NAME
      WRITE(6,2002)
      WRITE(6,2003)(T(M),E(M),EP(M),H(M),HP(M),M=1,JIN)
C
C        CALCULATE INPUT POWER THRU 10 DEGREES. DELTA(THETA) OF MUST
C        BE 0.25 DEG THRU 10 DEG.
C
      PINPS=0.0
      DO 11 J=5,41,4
        PINP=(E(J)+E(J-4))*(E(J)+E(J-4))*SIN((T(J)+T(J-4))/2.0*DTR)/4.
        PINPS=PINPS+PINP
11      CONTINUE
       WRITE(6,2011)PINPS
2011   FORMAT(1H0,'ASSUMING DELTA(THETA) OF INPUT PATTERN IS 0.25 DEG'
     &,' FOR 0 THRU 10 DEG,THE INPUT POWER THRU 10 DEG IS PROP TO',
     &F12.5,' SQUARE VOLTS')
C
C       CONVERT TO RADIANS AND REAL AND IMAG
C
      DO 45 J=1,JIN
```

```
45    T(J)=DTR*T(J)
      IF(IC2)50,50,60
50    DO 55 J=1,JIN
      TH=DTR*EP(J)
      EP(J)=E(J)*SIN(TH)
      E(J)=E(J)*COS(TH)
      TH=DTR*HP(J)
      HP(J)=H(J)*SIN(TH)
55    H(J)=H(J)*COS(TH)
60    CONTINUE
C
C        ESTABLISH DIFFERENTIAL PATTERN VALUES AT EACH INPUT POINT OF
C        PATTERN. VALUES ARE STORED IN VECTORS A & B AND ARE USED IN
C        EVALUATING THE COEFFICIENT INTEGRALS.
C
      A(1,1)=E(1)*T(1)
      A(1,2)=EP(1)*T(1)
      B(1,1)=H(1)*T(1)
      B(1,2)=HP(1)*T(1)
      DO 65 J=2,JIN
      DTH=T(J)-T(J-1)
      A(J,1)=DTH*E(J)
      A(J,2)=DTH*EP(J)
      B(J,1)=DTH*H(J)
      B(J,2)=DTH*HP(J)
65    CONTINUE
CC
C        FOR EACH COMBINATION OF MODE COMPONENT & INPUT THETA,DETERMINE
C        THE LEGENDRE POLY WEIGHTS USED IN EXPANSION. STORE IN NMAX
C        BY JIN MATRICES,F AND G.
C
      FMC=MCOMP
      DO 80 J=1,JIN
      Z=COS(T(J))
      DO 85 N=1,MCOMP
85    PM(N)=0.
      NC=NMAX+1
      CALL LEGEND(NC,MCOMP,Z,PM)
      DO 80 I=1,NMAX
      F(I,J)=FMC*PM(I)
      T1=I-MCOMP+1
      T2=I+1
      G(I,J)=T1*PM(I+1)-T2*Z*PM(I)
80    CONTINUE
C
C        EVALUATE COEFFICIENT INTEGRALS BY MULTIPLICATION OF F & G
C        MATRICES AND A & B MATRICES. ACOE & BCOE RETURNED FROM MULT.
C
      CALL MULT(JIN,NMAX,2,F,A,ACOE,0,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,G,B,ACOE,1,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,F,B,BCOE,0,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,G,A,BCOE,1,80,121,121,2,80,2)
C
C      NORMALIZE COEFFICIENTS AND COMPUTE POWER CONTRIBUTED FROM EACH
C        MODE.
C
      PIO2Z=(3.1415927/2.0)*0.002655
      PTOT=0.
      DO 95 N=1,NMAX
      PA(N)=0.
      PB(N)=0.
      FACT=1.
      IF(MCOMP)92,92,90
90    DO 91 M=1,MCOMP
      FF=(N-M+1)*(N+M)
91    FACT=FACT*FF
92    FF=2*N+1
      FACT=FACT/FF
      FF=2*N*(N+1)
      FACT=FACT*FF
      K=0
93    K=K+1
      BT=BCOE(N,K)
      AT=ACOE(N,K)
      BCOE(N,K)=BCOE(N,K)/FACT
      ACOE(N,K)=ACOE(N,K)/FACT
      PA(N)=PA(N)+AT*ACOE(N,K)*PIO2Z
      PB(N)=PB(N)+BT*BCOE(N,K)*PIO2Z
      IF(K-1)93,93,94
94    CONTINUE
      PTOT=PTOT+PA(N)+PB(N)
95    CONTINUE
C
C        WRITE SOME INPUTS ON DISK DATA SET
C
       WRITE(9,2223)NAMEJ
       WRITE(9,2224)NMAX,MCOMP
       WRITE(9,2223)NAME
C        OUTPUT COEFFICIENTS
C
C
      WRITE(6,2001)NAMEJ
      WRITE(6,2004)MCOMP
      PSUM=0
      DO 100 J=1,NMAX
      PA(J)=PA(J)/PTOT
      PB(J)=PB(J)/PTOT
      PSUM=PSUM+PA(J)+PB(J)
C
C        WRITE COEFFICIENTS ON DISK. COEFFICIENTS HAVE FAR-FIELD RADIAL
C        DEPENDENCE BUILT IN.
C
      WRITE(9,2222)J,ACOE(J,1),ACOE(J,2),BCOE(J,1),BCOE(J,2)
2222   FORMAT(I5,2E17.8,2X,2E17.8)
100   WRITE(6,2005)J,ACOE(J,1),ACOE(J,2),BCOE(J,1),BCOE(J,2),PA(J)
     &,PB(J),PSUM
      WRITE(6,2007)PTOT
C
C        COMPUTE FAR-FIELDS PATTERN OF SPHERICAL WAVE COEFF FOR
C        COMPARISON WITH INPUT PATTERN.
C
      READ(5,1002)JMAX,JIN
1011   FORMAT(1X,2I4)
      JO=0
      POUTS=0.0
      IC1=1
      IC2=-1
```

```
      WRITE(6,2008)
      WRITE(6,2002)
      DT=JMAX-1
      DT=180.0/DT
      J=1
      TH=0
      IF(MCOMP-1)170,180,170
170   DO 175  N=1,NMAX
      F(1,N)=0
175   G(1,N)=0
      GO TU 215
180   DO 190  N=1,NMAX
      FN=N*(N+1)
      F(1,N)=FN/2.0
190   G(1,N)=FN/2.0
      GO TO 215
C
C        ESTABLISH F & G MATRICES FOR OUTPUT THETAS;NMAX BY JOUT MATS.
C
200   FJ=J-1
      TH=FJ*DT
      Z=COS(DTR*TH)
      S=SIN(DTR*TH)
      DO 205  N=1,MCOMP
205   PM(N)=0
      CALL LEGEND(NC,MCOMP,Z,PM)
      DO 210  N=1,NMAX
      F(1,N)=FMC*PM(N)/S
      T1=N-MCOMP+1
      T2=N+1
      G(1,N)=T1*PM(N+1)-T2*Z*PM(N)
210   G(1,N)=G(1,N)/S
C
C        MULTIPLY F & G MATS AND ACOE & BCOE TO DETERMINE FIELD VALUES
C        AOUT & BOUT AT EACH OUTPUT ANGLE.
C
215   CALL MULT(NMAX,1,2,F,ACOE,AOUT,0,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,G,BCOE,AOUT,1,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,G,ACOE,BOUT,0,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,F,BCOE,BOUT,1,80,121,80,2,1,2)
C
      CALL VECTOR(AOUT(1,1),AOUT(1,2),EAMP,EPHI)
      CALL VECTOR(BOUT(1,1),BOUT(1,2),HAMP,HPHI)
      WRITE(6,2003)TH,EAMP,EPHI,HAMP,HPHI
       IF(J .EQ. 1)GO TO 274
       IF(J .GT. 1)GO TO 275
         POUT=(EAMP+EAMPT)*(EAMP+EAMPT)*SIN((TH+THT)/2.0*DTR)/4.0
         POUTS=POUTS+POUT
274      EAMPT=EAMP
         THT=TH
275   J=J+1
      IF(J-JIN)200,280,300
280   IF(J-JMAX)200,285,300
285   ISIGN=1
      TH=180.
      IF(MCOMP-1)370,380,370
370   DO 375  N=1,NMAX
      F(1,N)=0
375   G(1,N)=0
      GO TO 215
380   DO 290  N=1,NMAX
      ISIGN=-ISIGN
      FN=N*(N+1)*ISIGN
      F(1,N)=-FN/2.0
290   G(1,N)=FN/2.0
      GO TO 215
300   CONTINUE
C     GO TO 1
       WRITE(6,2012)POUTS
2012   FORMAT(1H ,'ASSUMING DELTA(OUTPUT THETA) IS 1.0 DEG,THE OUTPUT',
     &' POWER THRU 10 DEG IS',F12.5,' SQUARE VOLTS')
      STOP
3001  FORMAT(18A4)
C002  FORMAT(10I5)
C003  FORMAT(5F10.0)
2001  FORMAT(1H0,18A4)
2002  FORMAT(1H0,'      POLAR             E-PLANE            H-PLANE
     &/,'   ANGLE      VOLTS       DEG         VOLTS        DEG')
2003  FORMAT(F10.2,F12.6,F8.2,F13.6,F8.2)
2004  FORMAT(1H0'SPHERICAL WAVE COEFFICIENTS FOR AZIMUTHAL ORDER',I2//
     &20X,4HA(N),32X,4HB(N),27X,'  FRACTION OF TOTAL MODE POWER'/
     &5H    N,7X,4HREAL,13X,4HIMAG,15X,4HREAL,13X,4HIMAG,
     &13X,7HA MODES,6X,7HB MODES,5X,16HCUMULATIVE TOTAL)
2005  FORMAT(I5,2E17.8,2X,2E17.8,2X,2F14.5,F14.8)
2006  FORMAT(1H0,'INPUT PATTERN FOR AZIMUTHAL COMPONENT OF ORDER',I2)
2007  FORMAT(1H0,'    TOTAL MODE POWER    ',E15.8,' WATTS')
2008  FORMAT(1H0,'           FAR FIELD SUMMATION OF SPHERICAL MODES')
2009  FORMAT(F10.2,F10.6,F10.2,F10.6,F10.2)
2010  FORMAT(2I5)
      END
```

NATIONAL RADIO ASTRONOMY OBSERVATORY
GREEN BANK, WEST VIRGINIA

ADDENDUM
TO
ELECTRONICS DIVISION INTERNAL REPORT NO. 221

UPDATED DESCRIPTION OF USING THE JPL PHYSICAL-OPTICS SCATTERING PROGRAM

JAMES R. LYONS

AUGUST 1982

NUMBER OF COPIES: 150

UPDATED DESCRIPTION OF USING THE JPL PHYSICAL-OPTICS SCATTERING PROGRAM

James R. Lyons

I. Introduction

A field plotting routine, multiple-scattering capability, and several other features, have been added to the JPL program as described in EDIR No. 221 (September 1981). This memo is designed to briefly describe these additions and to provide the user with a convenient list of program inputs. Specifically, it is a revision of section III, Details of Using the Program, of report 221. It is suggested that the previous report be read before attempting to read and apply this memo to the program. Neither the basic structure nor the underlying theory of the program have been altered, but there are additional input parameters.

II. Added Program Features

A. Field plotting routine.

A subroutine, EPLOT, has been written which will plot the amplitude (dB) and phase of a particular component of an electric or magnetic field. The spherical wave expansion (SWE) program employs this subroutine to plot the E and H plane of the input field and of the far-field form of the spherical wave representation. The scattering (SCAT) program uses EPLOT to plot the polar, azimuthal, or radial component of the near-field of the incident pattern, and to plot the polar, azimuthal or Cartesian-Y component of the far-field form of the incident and scattered patterns. Only the first 30° of the far-field incident pattern can be plotted; for $\Theta(1) > 30°$, the plot and power of this pattern are skipped. Input parameters are used to choose the field component to be plotted.

In the output grid, the Y component of the electric field has the form:

$$E_Y(\Theta,\Phi) = E_\Theta(\Theta,\Phi) \cos \Theta \sin \Phi + E_\Phi(\Theta,\Phi) \cos \Phi.$$

The cos Θ term induces an azimuthal variation on E_Y which would destroy any beam circularity. Since our currently relevant problems involve circularly symmetric beams, the amplitude of this term has been neglected. E_Y then has the form

$$E_Y(\Theta,\Phi) = E_\Theta(\Theta,\Phi) \text{ sign } (\cos \Theta) \sin \Phi + E_\Phi(\Theta,\Phi) \cos \Phi.$$

Note also that $|\cos \Theta| \approx 1$ for most scattering situations.

B. Multiple scattering.

In some scattering problems, there exists several reflectors, so it becomes necessary to perform a spherical wave expansion on a scattered field and reflect this off yet another surface. SCAT can do this by writing the Y component of the far-field scattered field values it computes into a disk data set. The SWE progfam then reads in the values and determines the expansion. SCAT stores 31 values (for $\Delta\Theta = 1.0°$, this corresponds to 30°), either those on the "right side" of the beam or the mirror-image of those on the "left side" of the beam. In either case, the polar angle of the field values is translated such that the beam peak occurs at $\Theta = 0°$. Since only half the beam is stored, the entire beam must be assumed circular. If the scattered field is asymmetric with respect to its beam peak, individually expanding both beam halves will define the extremes within which the true beam must lie.

The following table shows the data flow for the two programs when scattering from multiple reflectors. The numbers are the data set reference numbers used in the READ and WRITE statements.

		READ	WRITE
First reflection	{SWE	Cards	9
	{SCAT	9	12
Subsequent reflections	{SWE	12	9
	{SCAT	9	12

Note: SCAT only stores the pattern values corresponding to the final Φ of the output grid.

C. Power calculations.

Relative power calculations are made for the same patterns that can be plotted by EPLOT. For the near-field of the incident pattern in SCAT, the power integral has the form

$$P = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\theta(\phi)} \left[|H_\theta(\theta,\phi)|^2 + |H_\phi(\theta,\phi)|^2 \right] \sin\theta \, d\theta \, d\phi .$$

For the far-field patterns in SCAT and the patterns in the SWE program, the integral is:

$$P = \int_0^{\theta_0} \left[|E_\theta(\theta,\Phi_0)|^2 + |E_\Phi(\theta,\Phi_0)|^2 \right] \sin\theta \, d\theta .$$

In the near-field integral, the power is calculated out to the reflector edge associated with each azimuthal angle. The cumulative power thru each integration grid is output along with the average value of the edge-angle of the integration grid. Power is also calculated for any section of the integration grid which lies beyond the reflector edge, thus allowing the spillover efficiency to be determined, if desired.

The far-field power integral can only be evaluated out to a constant polar angle. For the far-field incident pattern of SCAT, this angle is the average of the average edge-angles computed in the near-field case. For the scattered pattern in SCAT and the patterns in the SWE program, the angle thru which the power is calculated is READ in by the programs.

The power contribution between two polar angles on the grid is evaluated at the midpoint of the angles:

$$\Delta P = \frac{1}{2} [|E(\theta_1)|^2 + E(\theta_2)|^2] \cdot \sin\left(\frac{\theta_1 + \theta_2}{2}\right) \cdot \Delta\theta .$$

When the integration limit lies between grid values, ΔP is linearly scaled to the limit value. For the incident and SWE patterns, ΔP is doubled for each anulus because the pattern is actually that of a half-beam.

D. Field interpolation.

To accurately represent an input pattern, the SWE program requires values to be input at least every 0.25° or 0.5° in polar angle. Since most patterns are reasonably "smooth" thru the first 20 dB or so, points are input to the program every 1.0° or 2.0° and a linear interpolation is used to include points every 0.25° or 0.5°, respectively. To better represent the rounded beam peak, the first three interpolated values (i.e., .25, .5, .75, or .5, 1.0, 1.5) have a perturbation added to the interpolation.

The maximum number of points MAINOR can handle in computations is 121. Therefore, the maximum number of input points is 31 (e.g., 0° thru 30°, every 1.0°).

III. Input Variables and Their Formats

For both programs this section lists the input variables, gives a brief description of each variable, provides a concise table of inputs and their formats, and lists the output produced by the program.

A. SWE program.

1. Input variables descriptions:

Angles and phase are in degrees. All inputs read-in by MAIN.

NAMEJ	-	Alphanumeric: Use to identify program; $\leq$ 72 characters.
MCOMP	-	Order of azimuthal variation; usually = 1.
LMAX	-	Maximum mode order, $\leq$ 80; usually $\approx$ 70.
NAME	-	Alphanumeric: Use to identify reflector, pattern, etc.
JIN	-	Number of pattern points used in program, $\leq$ 121; = 4* (actual number of card inputs - 1) + 1.
IC1	-	If $\leq$ 0, convert input field from dB to volts.
IC2	-	If > 0, set input field phase to zero.
IC3	-	If > 0, compute pattern from equations inserted
IC4	-	If > 0, E and H plane patterns are identical; input only E plane.
NDISK	-	Pattern parameter: = 1 if values are to be input from cards; = 2 if values are read from disk data set (#12).
IPLANE	-	Specifies plane: = 1 if E plane power is calculated and E plane pattern is plotted; = 2 for H-plane.
IPLOT1	-	Input pattern print-out parameter: = 1 for a plot; = -1 for numerical values; = -11 for both; = 0 for neither.
IPLOT2	-	Output (SWE) pattern print-out parameter: Same as IPLOT1.
TEDGE	-	Polar angle thru which power flow is calculated.
T(J)	-	Polar angle of input pattern values.
E(J), H(J)	-	E, H-plane input pattern amplitude.
EP(J), HP(J)	-	E, H-plane input pattern phase.
JMAXO	-	= $180/\Delta\Theta + 1$, where $\Delta\Theta$ is desired output increment of the output (SWE) pattern; typically = 181.
JOUT	-	Number of output points starting with $\Theta = 0°$; typically = 91.

2. Table of inputs and their formats.

	Card No.	Variables	Format
	1	NAMEJ	18A4
	2	MCOMP, LMAX	2I5
	3	NAME	18A4
	4	JIN, IC1, IC2, IC3, IC4, NDISK	6I5
	5	IPLANE, IPLOT1, IPLOT2	3I5
	6	TEDGE	F10.5
If NDISK = 1	7	IC4≤0: T(1), E(1), EP(1), H(1), HP(1)	5F10.5
		IC4>0: T(1), E(1), EP(1)	3F10.5
	⋮		
	JW + 6	IC4≤0: T(JIN), etc.	5F10.5
		IC4>0: T(JIN), etc.	3F10.5
	JIN + 7	JMAXO, JOUT	2I5
If NDISK = 2	No cards: Data read from computer disk (#12)	NAMEJ	(18A4)
		T(1), E(1), EP(1)	(3F10.5)
		⋮	
		T(JIN), etc.	(3F10.5)
	7	JMAXO, JOUT	2I5

3. SWE program print-out.

- -- Two alphanumeric statements.
- -- Input pattern: plot or numberical values.
- -- Power in input pattern.
- -- Real and imaginary values of SWE coefficients.
- -- Fraction of total mode power in the coefficients for each mode order.
- -- Total coefficient mode power (not related to computed power in pattern).
- -- Output pattern (far-field of SWE): plot or numerical values.
- -- Power in output pattern.

B. SCAT Program.

1. Input variable descriptions.

All linear measures are in meters; all angles are in degrees.

Parentheses indicate which subroutine is reading in data.

(MAIN)

TITLE	-	Alphanumeric: describe reflector, wavelength, etc., $\leq$ 72 characters.
NDISK	-	disk writing parameter: = 1 if 1st half of scattered values are to be stored in a disk data set; = 2 for 2nd half of pattern; = 0 if no values are to be stored.
PC	-	phase constant = $2\pi/\lambda$
XT, YT, ZT	-	translation to expected phase center of scattered pattern.
ALPHA	-	used as a reflector rotation parameter when surface is specified analytically; set to 0.0 if not needed.
RHOØ	-	distance to the reflector at $\theta = 0°$.
THETAØ	-	used as an edge parameter when theta-edge is specified analytically; set to 0.0 if not needed.

- - - - - - - - - -

(SETUP)

JMAX	-	number of Θ values in output grid; $\leq$ 361.
TT1	-	initial Θ value.
DTT	-	Θ increment, usually = 1.0°.
KMAX	-	number of Φ values in output grid; $\leq$ 46.
PP1	-	initial Φ value.
DPP	-	Φ increment.
NG1	-	number of integration grids, $\leq$ 21.
MM(I)	-	number of θ values on Ith integration grid; $\leq$ 36.
T1	-	initial θ value.
DT	-	θ increment, typically between 0.2° and 1.0°.

NN(I) - number of ϕ values on Ith integration grid; $\leq$ 91.

P1 - initial ϕ value.

DP - ϕ increment, typically between 2° and 10°.

(Note: Subroutine EPLOT and the power calculations assume that DT and DP are constant within an integration grid, and that DTT is constant for the output grid.)

\- - - - - - - - -

(FIELDS)

NAME - alphanumberic: identify SWE program.

LMAX - maximum mode order.

MCOMP - order of azimuthal variation.

NAME - alphanumeric: identify reflector, pattern, etc.

J - 1 to LMAX.

A(J,1), A(J,2) - real and imaginary components of SWE "A" coefficients.

B(J,1), B(J,2) - real and imaginary components of SWE "B" coefficients.

\- - - - - - - - -

(MAIN)

IEDGE - reflector parameter used to determine if and how program obtains the reflector edge-values of θ. Cases:

$<$ 0 - calculate values using equations inserted in EDGEEQ . READ in 1 value, TEDGE1, to be used in equations.

= 0 - θ at edge is a constant. READ 1 value, TEDGE(1).

= 11 - edge will not be encountered on present integration grid. READ no values.

$>$ 0 - θ at edge is in tabular form and is read in for each ϕ on the integration grid. READ in NMAX values.

TEDGE(N) - edge value of θ. Read in either NMAX, 1, or no values.

\- - - - - - - -

(SURF)

ISURF - reflector parameter used to determine how $\rho(\theta,\phi)$, $\partial\rho/\partial\theta\ (\theta,\phi)$, and $\partial\rho/\partial\phi\ (\theta,\phi)$ are obtained. Cases:

< 0 ($\neq -2$) - ρ, $\partial\rho/\partial\theta$, $\partial\rho/\partial\phi$ all determined analytically from equations in SURF. READ no values.

$= -2$ - ρ determined from equations; derivatives numerically computed. READ no values.

$= 0$ - ρ is in tabular form and is a function of θ; derivatives numerically computed. READ MMAX values.

0 - ρ is in tabular form and is a function of θ and ϕ; derivatives numerically computed. READ MMAX x NMAX values.

F(M, N) - $\rho(\theta,\phi)$. READ either MMAX x NMAX, MMAX, or no values.

\- - - - - - - -

(MAIN)

NIPLOT - near-field incident pattern parameter:

$= 1, 2, 3$ - plot $H_\rho(\theta,\phi_0)$, $H_\theta(\theta,\phi_0)$, $H_\phi(\theta,\phi_0)$, respectively.

$= 0$ - no plot.

PHIPLT - ϕ_0 used with NIPLOT: ϕ_0 must be on present integration grid segment.

FIPLOT - far-field incident pattern parameter:

$= 1$ - plot $E_Y(\Theta,\Phi_0)$ = Y component.

$= 2, 3$ - plot $E_\Theta(\Theta,\Phi_0)$, $E_\Phi(\Theta,\Phi_0)$, respectively.

$= -1$ - numerical values listed.

$= 0$ - neither plot nor numerical values.

FSPLOT - far-field scattered pattern parameter. (Same as FIPLOT.)

TEDGEA - half-angle from beam center, thru which power in scattered pattern is computed.

2. Table of inputs and their formats.

A dash (-) in the Card No. column indicates that the card number is not known or is difficult to determine.

Card No.	Variables	Format
1	TITLE	18A4
2	NDISK	I5
3	PC, XT, YT, ZT, ALPHA, RHOØ, THETAØ	7F10.4
4	JMAX, TT1, DTT	I5, 2F10.2
5	KMAX, PP1, DPP	I5, 2F10.2
6	NG1	I5
7	MM(1), T1, DT	I5, 2F10.2
8	NN(1), P1, DP	I5, 2F10.2
⋮	⋮	⋮
2NG1 + 5	MM(NG1), T1, DT	I5, 2F10.2
2NG1 + 6	NN(NG1), P1, DP	I5, 2F10.2
NO CARDS:	NAME	(18A4)
Data	LMAX, MCOMP	(I5)
read	NAME	(18A4)
from	1, A(1,1), A(1,2), B(1,1), B(1,2)	(I5, 2E17.8, 2X, 2E17.8)
computer	⋮	⋮
disk	LMAX, A(LMAX,1), A(LMAX,2)	(I5, 2E17.8, 2X, 2E17.8)
(#9).	B(LMAX,1), B(LMAX,2)	
Read for each	IEDGE	I5
integration	TEDGE(N) (NMAX, 1, or no cards)	7(1X,F9.5)
grid.	ISURF	I5
Repeat NG1	F(M,N) (MMAX x NMAX, MMAX or no cards)	7(1X,F9.5)
times.	NIPLOT, PHIPLT, FIPLOT, FSPLOT	I5,F7.2,2I5
	TEDGEA	F10.4

3. SCAT program print-out.

-- 2 alphanumeric statements followed by propagation constant.

-- θ and ϕ for each integration grid segment.

For each integration grid.
- -- integration grid number and value of IEDGE.
- -- if edge is to be encountered, list of θ-edge and ρ-edge as functions of ϕ.
- -- integration grid number and value of ISURF.
- -- plot of near-field of incident pattern.
- -- near-field incident pattern power.

For each Φ.
- -- far-field of incident pattern: plot or numbers.
- -- far-field incident pattern power.

-- phase center translations, scale factor (set to 1.0).

For each Φ.
- -- far-field of scattered pattern: plot or numbers.
- -- far-field scattered pattern power.

C. Submitting a job and organization of JCL.

The SCAT and SWE programs have been run on the IBM 4341 computer located in Charlottesville. The operating system used is called Pandora; blocks of program code (e.g., subroutines) are stored under various Pandora member names (PMN^s).

To submit a job with the Pandora system, the SUBMIT command is used followed by the Pandora members making up the program and data. All the Pandora members (except EPLOT, which is just the plotting routine) have already been described in EDIR #221.

The submit command for the SWE program is SUBMIT_JCLSWE_MAINOR_EPLOT_LVM_DATASWE.

The JCL and the actual subroutines submitted are shown below. The PMN for a subroutine or block of subroutines is shown in the left-hand column.

PMN	JCL and Subroutines
	//SWAVE_ _JOB_(userI.D.),user name,MSGLEVEL=(2,0),
	CLASS=L,TIME=1
	/*ROUTE_ _PRINT_REMOTE1
	//_EXEC_FORTGCLG,ERROR=E,PARM.FORT=ID
	//FORT.SYSPRINT_DD_DUMMY
	//FORT.SYSIN_DD_*
MAINOR	MAIN
EPLOT	Subroutine EPLOT
LVM	Subroutines LEGEND, VECTOR, MULT
	/*
	//LKED.SYSPRINT_DD_DUMMY
	//GO.FT09F001_DD_DSN=user name.DATA,DISP=SHR
	//GO.FT12F001_DD_DSN=user name.DSCAT,DISP=SHR
	//GO.SYSIN_DD_*
DATASWE	DATASWE
	/*

A CLASS = L job uses a 216 K byte memory partition, double the size of a standard partition. The SWE program could be reduced to a standard partition by (carefully) reducing array sizes. If this is done, the corresponding dimentions in the CALL MULT statements in MAINOR must be changed along with the array dimensions in several subroutines.

For both SWE and SCAT jobs the TIME parameter is in (CPU) minutes.

The submit command for the SCAT program is

SUBMIT_JCLSCAT_MAIN1SC_EPLOT_SSURF_FINT_SFPVSALD_DATASCAT

The JCL and subroutine structure is as follows:

PMN	JCL and Subroutines

JCLSCAT	//SCAT_ _ _ _JOB_(userI.D.),user name,MSGLEVEL=(2,0),
	CLASS=0,TIME=user set
	- - - -
	- - - -
	- - - - Same as last 4 lines of JCLSWE
	- - - -

MAIN1SC	MAIN
EPLOT	Subroutine EPLOT
SSURF	Subroutine SURF
FINT	Subroutine FINT
SFPVSALD	Subroutines SETUP, FIELDS, PATHL, VECTOR, SPHANK, ADJUST, LEGEND, DIFF - - - - - - - - - - - - Same 5 lines of JCL as at end of member LVM - - - - - - - -
DATASCAT	DATASCAT /*

A CLASS = 0 job is an extra large partition of 880 K bytes. SCAT (between 300 and 400 K bytes) could be reduced to an L job by reducing the array dimensions of the output grid and the integration grid, if necessary, and by decreasing the number of allowable integration grids. Note that the array dimensions in several subroutines must be changed.

D. Creating and listing disk data sets.

The following JCL is used to create a sequential data set on the computer disk.

```
//CREATEDS_ _JOB_(userI.D.),user name,CLASS=Q,
     MSGLEVEL=1
/*ROUTE_ _PRINT_REMOTE1
//NEWFILE_ _ _EXEC_PGM=(,CATLG),DSN=user name.DATA
     or DSCAT,UNIT=3300,SPACE=(CYL,(1,1)),
        DCB=(RECFM=FB,LRECL=80,BLKSIZE=1600).
```

To list the contents of a data set:

```
//LIST_JOB_(userI.D.),user name,CLASS=Q,MSGLEVEL=1
/*ROUTE_ _PRINT_REMOTE1
//_EXEC_LIST
//SYSIN_DD_DSN=user name.DATA or DSCAT,DISP=SHR
```

Only the aobve JCL need be submitted to perform the desired task.

Acknowledgements

I thank Rick Fisher for his many helpful suggestions and constructive criticisms and Carolyn Dunkle for typing this report.

IV. Changed and Additional Program Code.

SWE PROGRAM

```
C        THIS IS A SPHERICAL WAVE EXPANSION PROGRAM-ORTHOGONALITY
C        TECHNIQUE.THE SPHERICAL WAVE COEFFICIENTS FOR THE M-TH
C        FOURIER COMPONENT OF A RADIATION PATTERN ARE COMPUTED.
C        THE EXPANSION MATCHES THE FAR-FIELD INPUT PATTERN WITH
C        THE THE FAR-FIELD FORM OF SPHERICAL WAVES.
C
      DIMENSION NAME(18),NAMEJ(18)
      DIMENSION PA(80),PB(80)
      DIMENSION T(121),E(121),EP(121),H(121),HP(121)
      DIMENSION F(80,121),G(80,121),PM(81)
      DIMENSION A(121,2),B(121,2)
      DIMENSION ACOE(80,2),BCOE(80,2)
      DIMENSION ACUT(1,2),BOUT(1,2)
C
      DTR=0.017453293
      DT2=DTR/2.0
C
C        SOME VARIABLE DEFINITIONS:
C
C     T,E,EP,H,HP=INPUT PATTERN PARAMETERS
C     JIN(<,=121)=# THETA INPUT VALUES
C     NMAX(<,=80) = MAXIMUM MODE ORDER
C     ACOE,BCOE = REAL & IMAG COMPONENTS OF TE AND TM WAVE COEFFICIENTS
C                 WRITTEN ON DISK
C     A,B = AMP & PHASE OF INPUT FIELDS VALUES * DELTA(THETA)
C     F,G =MATRICES (NMAX BY JIN OR NMAX BY JOUT OF VALUES RELATED TO
C          ASSOCIATED LEGENDRE POLYNOMIALS
C          G PROP. TO DERIVATIVE WRT THETA OF N-TH POLY.
C          F PROP. TO N-TH POLY.
C
C        INPUT ALPHANUMERIC STATEMENTS,PATTERN & PLOTTING PARAMETERS.
C
      READ(5,2223)NAMEJ
      WRITE(6,2001)NAMEJ
      READ(5,1002)MCOMP,NMAX
C
      READ(5,2223)NAME
       WRITE(6,2001)NAME
      READ(5,1002)JIN,IC1,IC2,IC3,IC4,NDISK
      READ(5,1002)IPLANE,IPLOT1,IPLOT2
      READ(5,2013)TEDGE
C
C        IF IC3>0,CALCULATE INPUT PATTERN PARAMETERS FROM EQUATION
C
       IF(IC3)113,113,133
C
C        IF IC3<,=0 AND IC4>0,READ IN ONLY T,E,EP. I.E.,PATTERN IS CIRC.
C
113   IF(IC4)153,153,163
163    CONTINUE
C
C        READ-IN PATTERN FROM DISK DATA SET (#12)
C
      IF(NDISK .EQ. 1)GO TO 110
       READ(12,2223)NAMEJ
       WRITE(6,2014)NAMEJ
       READ(12,2013)(T(J),E(J),EP(J),J=1,JIN,4)
      GO TO 115
```



```
110    CONTINUE
C
C         READ-IN PATTERMFROM CARDS
C
       READ(5,2013)(T(M),E(M),EP(M),M=1,JIN,4)
115   CONTINUE
C
C         CIRCULARLY SYMMETRIC BEAM
C
      DO 120 J=1,JIN,4
       H(J)=E(J)
       HP(J)=EP(J)
120   CONTINUE
      GO TO 130
C
153   READ(5,2015)(T(M),E(M),EP(M),H(M),HP(M),M=1,JIN,4)
C
C       FOR IC1 < OR = 0 CONVERT FROM DB TO VOLTS
C
130   IF(IC1)10,10,20
10    DO 15 J=1,JIN,4
      E(J)=10.0**(E(J)/20.0)
15    H(J)=10.0**(H(J)/20.0)
20    CONTINUE
C
C         INTERPOLATE  PATTERN VALUES
C
       JINM=JIN-4
       DO 167 J=1,JINM,4
        JP2=J+2
        DO 166 K=J,JP2
         FD4=FLCAT(K+1-J)/4.
          T(K+1)=T(J)+FD4*(T(J+4)-T(J))
          E(K+1)=E(J)+FD4*(E(J+4)-E(J))
          EP(K+1)=EP(J)+FD4*(EP(J+4)-EP(J))
          H(K+1)=H(J)+FD4*(H(J+4)-H(J))
          HP(K+1)=HP(J)+FD4*(HP(J+4)-HP(J))
C
C         ADD PERTURBATION
C
          IF(J .GT. 1)GO TO 166
           PERT=.25/FLOAT(K+J)
           E(K+1)=E(K+1)+PERT*(E(J)-E(J+4))
           H(K+1)=H(K+1)+PERT*(H(J)-H(J+4))
166       CONTINUE
167    CONTINUE
C
       GO TO 143
133    CONTINUE
       DELT=0.25
      DO 123 M=1,JIN
C
C          INSERT EQUATIONS FOR INPUT PATTERN HERE. NOTE ABOVE
C          INITIALIZATIONS.
C
          T(M)=(M-1)*CELT
          E(M)=EXP(-T(M)*T(M)/31.4)
          H(M)= E(M)
```

```
123   CONTINUE
143    CONTINUE
C
C       FOR IC2 > 0 NEGLECT PHASE
C
      IF(IC2)40,40,30
30    DO 35 J=1,JIN
      EP(J)=0.
35    HP(J)=0.
40    CONTINUE
C
C
       JPLT=(JIN-1)/4 +1
       DTT= T(5)-T(1)
      IF(IPLOT1 .EQ. 0)GO TO 160
      IF(IPLOT1 .EQ. -1)GO TO 150
C
C        LABEL PLOT : E OR H PLANE
C
       IF(IPLANE .EQ. 1)WRITE(6,2016)
       IF(IPLANE .EQ. -1)WRITE(6,2017)
       DO 140 J=1,JIN,4
        K=(J-1)/4 +1
C
C        PLOT DESIRED PLANE
C
      IF(IPLANE .EQ. 1)CALL EPLOT(K,E(J),EP(J),JPLT,DTT,T(1),T(J))
      IF(IPLANE .EQ. -1)CALL EPLOT(K,H(J),HP(J),JPLT,DTT,T(1),T(J))
140    CONTINUE
       IF(IPLOT1 .EQ. 1)GO TO 160
150    CONTINUE
C
C       PRINT CUT INPUT PATTERN
C
      WRITE(6,2006)MCOMP
      WRITE(6,2002)
      WRITE(6,2003)(T(M),E(M),EP(M),H(M),HP(M),M=1,JIN)
C
160    CONTINUE
C
C        CALCULATE INPUT POWER THRU THETA-EDGE. POWER IS CALCULATED
C         AT MIDPOINT OF CONSECUTIVE INPUT(!!) PATTERN VALUES
C         NOT INTERPOLATED VALUES.
C
      JPOW=INT(4*TEDGE) +1
      PINPS=0.0
      DO 11 J=5,JPOW,4
C
C        POWER IN E OR H PLANE
C
        IF(IPLANE .EQ. 1)PINP=(E(J)*E(J)+E(J-4)*E(J-4))*SIN((T(J)
     &+T(J-4))*DT2)
        IF(IPLANE .EQ. -1)PINP=(H(J)*H(J)+H(J-4)*H(J-4))*SIN((T(J)
     &+T(J-4))*DT2)
        PINPS=PINPS+PINP*DTT
11      CONTINUE
C
C        PRINT-CUT POWER RESULTS
```



```
C
        WRITE(6,2011)TEDGE,PINPS
C
C         CONVERT TO RADIANS AND REAL AND IMAG
C
      DO 45 J=1,JIN
45    T(J)=DTR*T(J)
      IF(IC2)50,50,60
50    DO 55 J=1,JIN
      TH=DTR*EP(J)
      EP(J)=E(J)*SIN(TH)
      E(J)=E(J)*COS(TH)
      TH=DTR*HP(J)
      HP(J)=H(J)*SIN(TH)
55    H(J)=H(J)*COS(TH)
60    CONTINUE
C
C        ESTABLISH DIFFERENTIAL PATTERN VALUES AT EACH INPUT POINT OF
C        PATTERN. VALUES ARE STORED IN VECTORS A & B AND ARE USED IN
C        EVALUATING THE COEFFICIENT INTEGRALS.
C
      A(1,1)=E(1)*T(1)
      A(1,2)=EP(1)*T(1)
      B(1,1)=H(1)*T(1)
      B(1,2)=HP(1)*T(1)
      DO 65 J=2,JIN
      DTH=T(J)-T(J-1)
      A(J,1)=DTH*E(J)
      A(J,2)=DTH*EP(J)
      B(J,1)=DTH*H(J)
      B(J,2)=DTH*HP(J)
65    CONTINUE
CC
C        FOR EACH COMBINATION OF MODE COMPONENT & INPUT THETA,DETERMINE
C        THE LEGENDRE POLY WEIGHTS USED IN EXPANSION. STORE IN NMAX
C        BY JIN MATRICES,F AND G.
C
      FMC=MCOMP
      DO 80 J=1,JIN
      Z=COS(T(J))
      DO 85 N=1,MCOMP
85    PM(N)=0.
      NC=NMAX+1
      CALL LEGEND(NC,MCOMP,Z,PM)
      DO 80 I=1,NMAX
      F(I,J)=FMC*PM(I)
      T1=I-MCOMP+1
      T2=I+1
      G(I,J)=T1*PM(I+1)-T2*Z*PM(I)
80    CONTINUE
C
C        EVALUATE COEFFICIENT INTEGRALS BY MULTIPLICATION OF F & G
C        MATRICES AND A & B MATRICES. ACOE & BCOE RETURNED FROM MULT.
C
      CALL MULT(JIN,NMAX,2,F,A,ACOE,0,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,G,B,ACOE,1,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,F,B,BCOE,0,80,121,121,2,80,2)
      CALL MULT(JIN,NMAX,2,G,A,BCOE,1,80,121,121,2,80,2)
```

```
C
C       NORMALIZE COEFFICIENTS AND COMPUTE POWER CONTRIBUTED FROM EACH
C         MODE.
C
      PIO2Z=(3.1415927/2.0)*0.002655
      PTOT=C.
      DO 95 N=1,NMAX
      PA(N)=0.
      PB(N)=0.
      FACT=1.
      IF(MCCMP)92,92,90
 90   DC 91 M=1,MCCMP
      FF=(N-M+1)*(N+M)
 91   FACT=FACT*FF
 92   FF=2*N+1
      FACT=FACT/FF
      FF=2*N*(N+1)
      FACT=FACT*FF
      K=0
 93   K=K+1
      BT=BCOE(N,K)
      AT=ACOE(N,K)
      BCOE(N,K)=BCOE(N,K)/FACT
      ACOE(N,K)=ACOE(N,K)/FACT
      PA(N)=PA(N)+AT*ACOE(N,K)*PIO2Z
      PB(N)=PB(N)+BT*BCCE(N,K)*PIO2Z
      IF(K-1)93,93,94
 94   CONTINUE
      PTOT=PTOT+PA(N)+PB(N)
 95   CONTINUE
C
C        WRITE A FEW PROGRAM INPUTS INTO A DISK DATA SET (#9)
C
       WRITE(9,2223)NAMEJ
       WRITE(9,1002)NMAX,MCCMP
       WRITE(9,2223)NAME
C
C        OUTPUT COEFFICIENTS
C
C     WRITE(6,2001)NAMEJ
      WRITE(6,2004)MCOMP
      PSUM=0
      DC 100 J=1,NMAX
      PA(J)=PA(J)/PTOT
      PB(J)=PB(J)/PTOT
      PSUM=PSUM+PA(J)+PB(J)
C
C        WRITE COEFFICIENTS ON DISK. COEFFICIENTS HAVE FAR-FIELD RADIAL
C        DEPENDENCE BUILT IN.
C
      WRITE(9,2222)J,ACOE(J,1),ACOE(J,2),BCOE(J,1),BCOE(J,2)
C
 100  WRITE(6,2005)J,ACCE(J,1),ACOE(J,2),BCOE(J,1),BCOE(J,2),PA(J)
     &,PB(J),PSUM
      WRITE(6,2007)PTOT
C
C        COMPUTE FAR-FIELDS PATTERN OF SPHERICAL WAVE COEFF FOR
C        COMPARISON WITH INPUT PATTERN.
```



```
C
      READ(5,1002)JMAX,JIN
      JO=0
      PCUTS=0.0
      IC1=1
      IC2=-1
      CT=JMAX-1
      DT=180.0/DT
      JPOW=INT(TEDGE/DT) +1
      J=1
      TH=0
      IF(MCCMP-1)170,180,170
 170  DO 175 N=1,NMAX
      F(1,N)=0
 175  G(1,N)=0
      GO TC 215
 180  DC 190 N=1,NMAX
      FN=N*(N+1)
      F(1,N)=FN/2.0
 190  G(1,N)=FN/2.0
      GO TO 215
C
C        ESTABLISH F & G MATRICES FOR OUTPUT THETAS;NMAX BY JOUT MATS.
C
 200  FJ=J-1
      TH=FJ*DT
      Z=COS(DTR*TH)
      S=SIN(DTR*TH)
      DC 205 N=1,MCCMP
 205  PM(N)=0
      CALL LEGEND(NC,MCOMP,Z,PM)
      DO 210 N=1,NMAX
      F(1,N)=FMC*PM(N)/S
      T1=N-MCOMP+1
      T2=N+1
      G(1,N)=T1*PM(N+1)-T2*Z*PM(N)
 210  G(1,N)=G(1,N)/S
C
C        MULTIPLY F & G MATS AND ACOE & BCOE TC DETERMINE FIELD VALUES
C        ACUT & BCUT AT EACH OUTPUT ANGLE.
C
 215  CALL MULT(NMAX,1,2,F,ACOE,AOUT,0,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,G,BCOE,AOUT,1,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,G,ACOE,BOUT,0,80,121,80,2,1,2)
      CALL MULT(NMAX,1,2,F,BCOE,BOUT,1,80,121,80,2,1,2)
C
      CALL VECTCR(ACUT(1,1),AOUT(1,2),EAMP,EPHI)
      CALL VECTCR(BCUT(1,1),BOUT(1,2),HAMP,HPHI)
C
      IF(J .GT. JPLT)GC TC 270
      IF((IPLCT2 .EQ. C) .CR. (IPLCT2 .EQ. -1))GC TO 270
C
C        LABEL & PLCT CUTPUT PATTERN
C
      IF((J .EQ. 1) .AND. (IPLANE .EQ. 1))WRITE(6,2018)
      IF((J .EQ. 1) .AND. (IPLANE .EQ. -1))WRITE(6,2019)
      IF(IPLANE .EQ. 1)CALL EPLOT(J,EAMP,EPHI,JPLT,CT,C.0,TH)
      IF(IPLANE .EQ. -1)CALL EPLOT(J,HAMP,HPHI,JPLT,DT,0.0,TH)
```

```
G LEVEL  21                MAIN            DATE = 82211        11/15/21

 270     CONTINUE                                                          03240000
 C                                                                         03250000
         IF(J .EQ. 1)GO TO 274                                             03260000
         IF(J .GT. JPCW)GO TO 276                                          03270000
 C                                                                         03271000
 C         CALCULATE POWER IN OUTPUT PATTERN                               03272000
 C                                                                         03273000
        IF(IPLANE .EQ. 1)POUT=(EAMP*EAMP+EAMPT*EAMPT)*SIN((TH+THT)*DT2)    03280000
        IF(IPLANE .EQ. -1)PCUT=(HAMP*HAMP+HAMPT*HAMPT)*SIN((TH+THT)*DT2)   03285000
           POUTS=POUTS+PCUT*DT                                             03290000
 274       EAMPT=EAMP                                                      03300000
           HAMPT=HAMP                                                      03305000
           THT=TH                                                          03310000
 276     CONTINUE                                                          03315000
         E(J)=EAMP                                                         03320000
         EP(J)=EPHI                                                        03330000
         H(J)=HAMP                                                         03340000
         HP(J)=HPHI                                                        03350000
         T(J)=TH                                                           03360000
 275    J=J+1                                                              03370000
        IF(J-JIN)200,280,300                                               03380000
 280    IF(J-JMAX)200,285,300                                              03390000
 285    ISIGN=1                                                            03400000
        TH=180.                                                            03410000
        IF(MCOMP-1)370,380,370                                             03420000
 370    DO 375 N=1,NMAX                                                    03430000
        F(1,N)=0                                                           03440000
 375    G(1,N)=0                                                           03450000
        GO TO 215                                                          03460000
 380    DO 290 N=1,NMAX                                                    03470000
        ISIGN=-ISIGN                                                       03480000
        FN=N*(N+1)*ISIGN                                                   03490000
        F(1,N)=-FN/2.0                                                     03500000
 290    G(1,N)=FN/2.0                                                      03510000
        GO TO 215                                                          03520000
 300    CONTINUE                                                           03530000
        IF((IPLOT2 .EQ. 0) .OR. (IPLOT2 .EQ. 1))GO TO 310                  03540000
        WRITE(6,2008)                                                      03550000
        WRITE(6,2002)                                                      03560000
 C                                                                         03561000
 C         PRINT-OUT PATTERN VALUES                                        03562000
 C                                                                         03563000
        WRITE(6,2003)(T(J),E(J),EP(J),H(J),HP(J),J=1,JPLT)                 03570000
 310    CONTINUE                                                           03580000
         WRITE(6,2012)TEDGE,POUTS                                          03590000
        STOP                                                               03600000
 1002    FORMAT(6I5)                                                       03610000
 1011    FORMAT(1X,2I4)                                                    03620000
 2001   FORMAT(1H0,18A4)                                                   03630000
 2002   FORMAT(1H0,'     POLAR            E-PLANE            H-PLANE       ', 03640000
      &/,'   ANGLE      VOLTS      DEG         VOLTS       DEG')           03650000
 2003   FORMAT(F10.2,F12.6,F8.2,F13.6,F8.2)                                03660000
 2004   FORMAT(1H0'SPHERICAL WAVE COEFFICIENTS FOR AZIMUTHAL ORDER',I2//   03670000
      &20X,4HA(N),32X,4HB(N),27X,'  FRACTION OF TOTAL MODE POWER'/         03680000
      &5H    N,7X,4HREAL,13X,4HIMAG,15X,4HREAL,13X,4HIMAG,                 03690000
      &13X,7HA MODES,6X,7HB MODES,5X,16HCUMULATIVE TOTAL)                  03700000
 2005   FORMAT(I5,2E17.8,2X,2E17.8,2X,2E14.5,F14.8)                        03710000
 2006   FORMAT(1H0,'INPUT PATTERN FOR AZIMUTHAL COMPONENT OF ORDER',I2)    03720000
```

```
G LEVEL  21                MAIN            DATE = 82211        11/15/21

 2007   FORMAT(1H0,'   TOTAL MODE POWER   ',E15.8,' WATTS')                03730000
 2008   FORMAT(1H0,'           FAR FIELD SUMMATION OF SPHERICAL MODES')    03740000
 2009   FORMAT(F10.2,F10.6,F10.2,F10.6,F10.2)                              03750000
 2011    FORMAT(1H0,'THE INPUT POWER THRU ',F6.2,' DEGREES IS ',F6.4,      03760000
       &' SQUARE VOLTS')                                                   03770000
 2012    FORMAT(1H0,'THE OUTPUT POWER THRU ',F6.2,' DEGREES IS ',F6.4,     03780000
       &' SQUARE VOLTS')                                                   03790000
 2013   FORMAT(3F10.5)                                                     03800000
 2014   FORMAT('0','RADIATION SOURCE: ',18A4)                              03810000
 2015   FORMAT(5F10.5)                                                     03820000
 2016   FORMAT('0',53X,'FAR-FIELD INPUT PATTERN',/,14X,'E-PLANE')          03830000
 2017   FORMAT('0',53X,'FAR-FIELD INPUT PATTERN',/,14X,'H-PLANE')          03840000
 2018   FORMAT('1',53X,'FAR-FIELD OUTPUT PATTERN',/,14X,'E-PLANE')         03850000
 2019   FORMAT('1',53X,'FAR-FIELD OUTPUT PATTERN',/,14X,'H-PLANE')         03860000
 2222    FORMAT(I5,2E17.8,2X,2E17.8)                                       03870000
 2223    FORMAT(18A4)                                                      03880000
        END                                                                03890000
```

```
      SUBROUTINE EPLOT(J,EPAMP,EPPHI,JMAX,DTT,TT1,TC)
C
C        THIS SUBROUTINE PLOTS THE AMPLITUDE & PHASE OF THE FIELCS IN
C        MAINCR & MAINISC.EPLOT IS CALLED FOR EACH FIELD POINT. THE AMP
C        RANGE IS -46 DB TC +2 DB,WITH .4 DB INCREMENT;PHASE RANGE IS
C        -180 TC +180 DEG,INCREMENT OF 3 DEG. PLCTTING IS DONE BY
C        WRITING AMP & PHASE SYMBOLS (X&O) INTO APPRCPRIATE LOCATICNS
C        IN AN ARRAY,IARRAY,AND THEN PRINTING THE ARRAY ACRCSS THE PAGE
C
C         J=INDEX OF FIELC POINT
C         EPAMP,EPPHI=AMP & PHASE CF FIELD POINT
C         JMAX=# CF FIELD PCINTS
C         DTT=PCLAR ANGLE INCREMENT (DEG)
C         TT1=PCLAR ANGLE FOR J=1
C         TC=POLAR ANGLE CORRESPONCING TC PRESENT J
C
      INTEGER IARRAY(121)
      INTEGER EX,EYE ,BLANK,CASH,OH,    PLUS
      CATA EX/'X'/,  OH/'O'/,BLANK/' '/,DASH/'-'/,  EYE/'I'/,PLUS/'+'/
      IF(J .NE. 1)GC TC 815
C
C        INITIALIZE PLCTTING PARAMETERS DURING FIRST CALL
C
      CTR=C.017453293
      NCDASH=0
      NPRINT=0
C
C        INITIALIZE PEAK AMP
C
      PKAMP=-47.0
      CTTPLT =1C.0*DTT
      IDTT=INT(CTT)
       IF(IDTT .EQ. 0)IDTT=1
C
C        INITIALIZE IARRAY
C
      DC 805 L=1,121
805    IARRAY(L)=BLANK
C
C        PRINT PRELIMINARY PLOT LABELS
C
      WRITE(6,2107)
      WRITE(6,2102)
      WRITE(6,2104)
815    IF(INT(TC+.C1) .LT. INT(NCDASH*DTTPLT+TT1+.C1))GO TC 86C
C
C        WHEN ABOVE CONDITION IS MET,PRINT A REFERENCE LINE OF DASHES
C
        NCDASH=NCDASH+1
       NPRINT=1
       DO 835 M=1,121
835     IARRAY(M)=DASH
860    CONTINUE
      IF(JMAX .LT. 61)GO TC 863
        IF(J .NE. (JMAX+1)/2)GC TO 863
C
C        PRINT AMP & PHASE SCALE GRID MARKS IN CENTER CF PLCT. SKIP FOR
C        J < 61
C
        DO 861 M=1,121,10
861       IARRAY(M)=EYE
863     CONTINUE
C
C         CONVERT AMP FROM VOLTS TO DB
C
        IF(EPAMP .GE. 0.001)EAMPDB=20.*ALOG10(EPAMP)
        IF(EPAMP .LT. 0.001)EAMPDB=-50.0
C
C         ASSIGN PEAK AMP
C
        IF(EAMPDB .GT. PKAMP)PKAMP=EAMPDB
C
C         DETERMINE ARRAY LOCATIONS
C
       SAMP=2.5*(EAMPDB + 46.0)
       SPHASE=.3333333*(EPPHI+180.)
       NAP=                 INT(SAMP+.5)
       NPP=                   INT(SPHASE+.5)
C
       IF(NPP .NE. NAP)GO TC 830
C
C         IF AMP & PHASE LOCATIONS COINCIDE,PRINT A '+'
C
       IARRAY(NPP)=PLUS
       GO TO 833
C
C         ASSIGN AMP & PHASE SYMBOLS TO IARRAY
C
830    IF((NAP .GE. 0) .AND. (NAP .LE. 120))IARRAY(NAP+1)=EX
       IARRAY(NPP+1)=OH
C
C         SUPRESS PHASE SYMBOLS FOR AMP < -46 DB
C
       IF(NAP .LT. 0)IARRAY(NPP+1)=BLANK
833    DO 850 M=1,IDTT
         IF(NPRINT .NE. 1)GC TO 840
C
C         PRINT ANGLE,DASH,REFERENCE LINE WITH AMP & PHASE,AND DASH
C
          WRITE(6,2103)TO,DASH,IARRAY,DASH
845       NPRINT=0
        GC TO 855
C
C         PRINT DASH,AMP & PHASE,AND DASH
C
840    WRITE(6,21C1)CASH,IARRAY,DASH
855     IF(M .GT. 1)GO TO 850
        DC 820 I=1,121
820      IARRAY(I)=BLANK
850    CONTINUE
      IF(JMAX .LT. 61)GO TO 870
C
C         PRINT AMP & PHASE SCALE VALUES IN CENTER CF PLOT
C
       IF(J .EQ. (JMAX-1)/2)WRITE(6,2113)
       IF(J .EQ. (JMAX+3)/2)WRITE(6,2114)
```

```
870     CONTINUE
C
C         FINISH PLOT
C
       IF(J .NE. JMAX)RETURN
C
C         PRINT OUTPUT PLOT LABELS AND PEAK AMP
C
      WRITE(6,2104)
      WRITE(6,2110)
      WRITE(6,2109)
      WRITE(6,2112) PKAMP
      RETURN
2101    FORMAT(8X,1A1,121A1,1A1)
2102  FORMAT(' ',6X,'46.0      42.0      38.0      34.0      30.0'
     &,'      26.0      22.0      18.0      14.0      10.0       6.0'
     &,'       2.0    0  +2.0')
2103      FORMAT(4X,F4.0,123A1)
2104  FORMAT(' ',7X,'*I*********I*********I*********I*********I*****',
     &'****I*********I*********I*********I*********I*********I******',
     &'***I*********I*')
2107  FORMAT(' ',7X,'                                                      ',
     &'     X: AMPLITUDE,(-)DB')
2109  FORMAT(8X,'                                                        ',
     &' O: PHASE,DEGREES')
2110  FORMAT(7X,'-180       -150       -120        -90        -60       ',
     &' -30         0         30         60         90        120       ',
     &'150        180')
2112  FORMAT(' ',107X,'PEAK AMPLITUDE=',F6.2,'DB')
2113  FORMAT('+',8X,'DB       42         38         34          30       '
     &,'26              18         14         10          6       ',
     &'2         +2')
2114  FORMAT('+',8X,'DEG      -150       -120        -90        -60      '
     &,'-30             30         60         90        120       1',
     &'50      180')
      END
```



```
      SUBROUTINE MULT(M,N,K,A,B,C,IC,NA,MA,NB,MB,NC,MC)
C
C       THIS SUB. MULTIPLIES 2 FULL MATRICES
C
      DIMENSION A(NA,MA),B(NB,MB),C(NC,MC)
      IF(IC)10,20,30
10    DO 15 I=1,N
      DO 15 J=1,K
      DO 15 L=1,M
15    C(I,J)=C(I,J)-A(I,L)*B(L,J)
      RETURN
20    DO 25 I=1,N
      DO 25 J=1,K
      C(I,J)=0.0
      DO 25 L=1,M
25    C(I,J)=C(I,J)+A(I,L)*B(L,J)
      RETURN
30    DO 35 I=1,N
      DO 35 J=1,K
      DO 35 L=1,M
35    C(I,J)=C(I,J)+A(I,L)*B(L,J)
      RETURN
      END
```

THIS SUBROUTINE WAS INADVERTENTLY LEFT OUT OF THE FIRST REPORT.

```
G LEVEL  21                MAIN              DATE = 82211          11/19/25
                        SCAT PROGRAM
C         MAIN PROGRAM FOR SCATTERING FROM  ASYMMETRICAL SUBREFLECTORS
C           ASSUME SMOOTH PERFECTLY CONDUCTING SURFACE OF ARBITRARY
C           SHAPE
C
C         PROGRAM ORIGINALLY DUE TO A. LUDWIG ,FEB. 1970
CCCCCCCC
CCCCCCCC
      COMMON/GRID1/SIT( 36),COT( 36),SIP( 91),COP( 91),T(36),P( 91)
      COMMON/GRID2/SITT(361),COTT(361),SIPP(46),COPP(46),TT(361),PP(46)
      DIMENSION TITLE(18),RHOEDG(91),TTOUT(31),TPHI(31),TAMP(31)
      DIMENSION F(36,91),FT(36,91),FP(36,91),GAM(36,91),TEDGE(91)
      INTEGER TITLE,FIPLOT,FSPLOT
C
      COMPLEX HR(36, 91),HT(36, 91),HP(36, 91)
      COMPLEX ETT(361,46),EPP(361,46)
      COMPLEX ETTC,EPPC,EPLANE
      COMPLEX A(36,91,3)
      COMPLEX STOT(3)
      COMPLEX T1,T2,T3
      COMPLEX TET,TEP
      EQUIVALENCE (A(1,1,1),HR),(A(1,1,2),HT),(A(1,1,3),HP)
      EQUIVALENCE (GAM,FT)
C
C     T,P = THETA & PHI OF INTEGRATION GRID
C     TT,PP = BIGTHETA & BIGPHI OF OUTPUT GRID
C     COMMON PARAMETERS = SIN,COS OF ABOVE ANGLES
C     F,FT,FP = RHO AND ITS DERIVATIVES WRT TO THETA & PHI
C     HR,HT,HP = INCIDENT MAGNETIC FIELDS COMPONENTS
C     A = VECTOR RELATED TO SURFACE CURRENTS
C     STOT = VALUE OF RADIATION INTEGRAL AS RETURNED FROM FINT
C     GAM = PATHLENGTH TERM (A FUNCTION OF BOTH GRIDS)
C     TEDGE = POLAR ANGLE VALUES WHICH SPECIFY REFLECTOR EDGE AS A
C             FUNCTION OF PHI
C     ETT,EPP = SCATTERED FIELDS VALUES ON THE OUTPUT GRID
C     JMAX,KMAX = # BIGTHETA,#BIGPHI POINTS ON OUTPUT GRID
C     MMAX,NMAX = # THETA,#PHI POINTS ON PRESENT INTEGRATION GRID
C     NG1 = # INTEG GRIDS
C     ETTC,EPPC = FIELDS VALUES AT A SPECIFIC OUTPUT GRID POINT
C
      READ(5,1001)TITLE
      WRITE(6,2001)TITLE
      READ(5,1003)NDISK
C
C        WRITE INTO DISK DATA SET
C
      IF(NDISK .NE. 0)WRITE(12,1001)TITLE
C
C       INPUT PHASE TRANSLATIONS AND REFLECTOR PARAMETERS
C
      READ(5,1002)PC,XT,YT,ZT,ALPHA,RHOO,THETAO
21    SCALE=1.0
22    CONTINUE
      DTR=0.017453293
      WRITE(6,2003)PC
C
C       INPUT GRID DATA AND ESTABLISH OUTPUT GRID(#2) AND
C        FIRST INTEGRATION GRID(#1)
C
```

```
G LEVEL  21                MAIN              DATE = 82211          11/19/25
      CALL SETUP(NG1,I,JMAX,KMAX,MMAX,NMAX)
C
C     WRITE(6,2002)TITLE
C
C       INPUT INCIDENT FIELD DATA
      CALL FIELDS
C
C       BEGIN LOOP FOR INTEGRATION GRID SEGMENTS
      DO 90 J=1,JMAX
      DO 90 K=1,KMAX
      ETT(J,K)=(0.0,0.0)
90    EPP(J,K)=(0.0,0.0)
C
C         INITIALIZE PARAMETERS USED TO DETERMINE AVERAGE THETA-EDGE
C
      TEDGET=0.0
      NGEDGE=0
C
100   CONTINUE
C
      DT=(T(2)-T(1))/DTR
      DP=ABS(P(NMAX)-P(1))/DTR/360.0
      DTT=(TT(2)-TT(1))/DTR
      THMAX=T(MMAX)/DTR
C
C       INPUT IEDGE
C
      READ(5,1003)IEDGE
      WRITE(6,2021)I,IEDGE
C
C         INITIALIZE TEDGE
C
       DO 24 N=1,NMAX
24      TEDGE(N)=0.
23    CONTINUE
C
C         CHECK IF REFLECTOR EDGE IS WITHIN PRESENT INTEG GRID
C
      IF(IEDGE .EQ. 11)GO TO 28
       WRITE(6,2022)
C
C         IF EDGE IS PRESENT,DETERMINE TEDGE. OBTAIN TEDGE ACCORDING TO
C         VALUE OF IEDGE.
C
      IF(IEDGE)25,27,26
25     CONTINUE
       READ(5,1004)TEDGE1
       CALL EDGEEQ(NMAX,THETAO,TEDGE1,TEDGE)
       GO TO 29
26     CONTINUE
       READ(5,1004)(         TEDGE(N),N=1,NMAX)
       GO TO 29
27     CONTINUE
       READ(5,1004)          TEDGE(1)
C
C         PRINT-OUT CONSTANT THETA-EDGE
C
       IF(IEDGE .EQ. 0)WRITE(6,2019)(JJ,TEDGE(JJ),JJ=1,1)
```

```
G LEVEL  21                  MAIN              DATE = 82211          11/19/25
       DO 30 N=2,NMAX                                                   00970000
        TEDGE(N)=TEDGE(1)                                               00980000
 30    CONTINUE                                                         00990000
 C                                                                      00991000
 C        PRINT-OUT THETA-EDGE AS A FUNCTION OF PHI                     00992000
 C                                                                      00993000
 29    IF(IEDGE .NE. 0)WRITE(6,2019)(JJ,TEDGE(JJ),JJ=1,NMAX)            01000000
       WRITE(6,2018)                                                    01010000
 C                                                                      01011000
 C        CONVERT THETA-EDGE TO RADIANS;CALCULATE RHO-EDGE              01013000
 C                                                                      01014000
       DO 31 N=1,NMAX                                                   01020000
        TEDGE(N)=TEDGE(N)*0.01745329                                    01030000
 31     RHOEDG(N)=RHCO/(SIN(TEDGE(N))*COP(N)+ COS(TEDGE(N)))            01040000
 C                                                                      01041000
 C        OUTPUT RHO-EDGE                                               01050000
 C                                                                      01060000
 C        PRINT-OUT RHO-EDGE AS A FUNCTION OF PHI                       01061000
 C                                                                      01062000
       WRITE(6,2019)(JJ,RHOEDG(JJ),JJ=1,NMAX)                           01070000
 28       CONTINUE                                                      01080000
 C                                                                      01090000
 C        FIND AVERAGE-IN-PHI OF THETA-EDGE                             01100000
 C                                                                      01110000
        TEDGEA=C.                                                       01120000
        DC 405 N=1,NMAX                                                 01130000
         TEDGEA=TEDGEA + TEDGE(N)                                       01140000
 405     IF(IEDGE .EQ. 1)TEDGE(N)=THMAX                                 01150000
        TEDGEA=TEDGEA/FLOAT(NMAX)/0.017453293                           01160000
 C                                                                      01161000
 C        IF EDGE NOT TO BE ENCOUNTERED THIS GRID,SET TEDGEA TO THETAMAX 01162000
 C                                                                      01163000
        IF(TEDGEA .EQ. 0)TEDGEA=THMAX                                   01170000
 C                                                                      01180000
 C       ESTABLISH SURFACE PARAMETERS(I.E. RHO &ITS DERIVS.) ON         01190000
 C         INTEGRATION GRID                                             01200000
 C                                                                      01201000
       CALL SURF(I,MMAX,NMAX,F,FT,FP,IEDGE,TEDGE,RHOO,ALPHA)            01210000
       DC 410 M=1,MMAX                                                  01220000
       DC 410 N=1,NMAX                                                  01230000
       F(M,N)=PC*F(M,N)                                                 01240000
 410    CONTINUE                                                        01250000
 C                                                                      01260000
 C       ESTABLISH INCIDENT FIELDS ON INTEGRATION GRID BY EVALUATING    01270000
 C         NEAR-FIELD OF SWE                                            01280000
 C                                                                      01290000
       CALL FIELD1(MMAX,NMAX,HR,HT,HP,F)                                01300000
 C                                                                      01310000
 C        READ IN PLOTTING PARAMETERS                                   01320000
 C                                                                      01330000
       READ(5,1005)NIPLOT,PHIPLT,FIPLOT,FSPLOT                          01340000
 C                                                                      01350000
 C        PLOT NEAR-FIELD OF INCIDENT PATTERN                           01360000
 C                                                                      01370000
       TT1=T(1)/DTR                                                     01390000
       DC 430 M=1,MMAX                                                  01400000
       TC=T(M)/DTR                                                      01410000
       IF(M .NE. 1)GC TC 300                                            01420000

G LEVEL  21                  MAIN              DATE = 82211          11/19/25
 C                                                                      01421000
 C        LABEL PLOT WITH FIELD COMPONENT NAME                          01423000
 C                                                                      01424000
         IF(NIPLOT .EQ. 1)WRITE(6,2100)PHIPLT                           01430000
         IF(NIPLOT .EQ. 2)WRITE(6,2101)PHIPLT                           01440000
         IF(NIPLOT .EQ. 3)WRITE(6,2102)PHIPLT                           01450000
 300   CONTINUE                                                         01460000
        DO 440 N=1,NMAX                                                 01470000
 C                                                                      01471000
 C        CONVERT REAL & IMAG FIELD VALUES TO AMP & PHASE               01472000
 C                                                                      01473000
         A1=REAL(HR(M,N))                                               01480000
         A2=AIMAG(HR(M,N))                                              01490000
         A3=REAL(HT(M,N))                                               01500000
         A4=AIMAG(HT(M,N))                                              01510000
         A5=REAL(HP(M,N))                                               01520000
         A6=AIMAG(HP(M,N))                                              01530000
       CALL VECTCR(A1,A2,HRAMP,HRPHI)                                   01540000
       CALL VECTCR(A3,A4,HTAMP,HTPHI)                                   01550000
       CALL VECTOR(A5,A6,HPAMP,HPPHI)                                   01560000
       IF(NIPLOT .EQ.  0)GC TO 310                                      01570000
 C                                                                      01571000
 C        PLOT FOR PHI=PHIPLT                                           01572000
 C                                                                      01573000
       IF(ABS(P(N)/DTR-PHIPLT) .GE. 0.01)GO TO 310                      01580000
 C                                                                      01581000
 C        PLOT DESIRED FIELD COMPONENT                                  01582000
 C                                                                      01583000
         IF(NIPLOT .EQ. 1)CALL EPLOT(M,HRAMP,HRPHI,MMAX,DT,TT1,TO)      01590000
         IF(NIPLOT .EQ. 2)CALL EPLOT(M,HTAMP,HTPHI,MMAX,DT,TT1,TO)      01600000
         IF(NIPLOT .EQ. 3)CALL EPLOT(M,HPAMP,HPPHI,MMAX,DT,TT1,TO)      01610000
 310   CONTINUE                                                         01620000
 440    CONTINUE                                                        01630000
 430    CONTINUE                                                        01640000
 C                                                                      01650000
 C        COMPUTE NEAR-FIELD INCIDENT POWER                             01660000
 C                                                                      01670000
       TDIFF=.5*DT*DTR                                                  01680000
       IF(I .GT. 1)GC TC 305                                            01690000
 C                                                                      01690100
 C        INITIALIZE CUMULATIVE POWER TERMS. TPOW IS TOTAL POWER,POW IS 01690200
 C        POWER CUT TO EDGE                                             01690300
 C                                                                      01690400
         TPOWI=0.0                                                      01691000
         POWI=0.C                                                       01692000
 305   CONTINUE                                                         01693000
 C                                                                      01695000
 C        INITIALIZE POWER VARIABLES FOR EACH INTEGRATION GRID SEGMENT  01696000
 C                                                                      01697000
         POWX=0.C                                                       01700000
         TPOWX=0.0                                                      01710000
 C                                                                      01711000
 C        SUM POWER CONTRIBUTION FROM EACH DELTA-PHI                    01712000
 C                                                                      01713000
       DC 340 N=2,NMAX                                                  01720000
 C                                                                      01721000
 C        SUM POWER CONTRIBUTONS FROM EACH DELTA-THETA (TO THETA-EDGE)  01722000
 C                                                                      01723000
```

```
      DO 330 M=1,MMAX                                                   01730000
        A3=REAL(HT(M,N))                                                01740000
        A4=AIMAG(HT(M,N))                                               01750000
        A5=REAL(HP(M,N))                                                01760000
        A6=AIMAG(HP(M,N))                                               01770000
      CALL VECTCR(A3,A4,HTAMP,HTPHI)                                    01780000
      CALL VECTCR(A5,A6,HPAMP,HPPHI)                                    01790000
C                                                                       01791000
C        SUM SQUARE OF THETA & PHI FIELD COMPONENTS                     01792000
C                                                                       01793000
       EEM=HTAMP*HTAMP+HPAMP*HPAMP                                      01800000
       IF(M .EQ. 1)GO TO 325                                            01810000
C                                                                       01811000
C        POWER AT MIDPOINT OF ANULUS;MULTIPLY BY WIDTH OF ANULUS        01812000
C                                                                       01813000
       POWM=(EEMM1+EEM)*SIN(T(M)-TDIFF)                                 01820000
       TPCWX=TPCWX+PCWM*DT                                              01830000
C                                                                       01831000
C        TAKE AVERAGE OF PAST & PRESENT THETA-EDGE VALUES               01832000
C                                                                       01833000
       TEAVG=.5*(TEDGE(N)+TEDGE(N-1))                                   01840000
       IF(T(M) .LT. TEAVG)GO TO 320                                     01850000
C                                                                       01851000
C        SCALE POWER CONTRIBUTIONS AT EDGE IF TEAVG LIES BETWEEN THETAS 01852000
C                                                                       01853000
        SCALH=(TEAVG-T(M-1))/(T(M)-T(M-1))                              01860000
        IF(SCALH .LE. .01)GO TO 325                                     01870000
        POWX=POWX+SCALH*PCWM*DT                                         01880000
        GO TO 325                                                       01890000
 320   CONTINUE                                                         01900000
       POWX=POWX+POWM*DT                                                01910000
 325   CONTINUE                                                         01915000
       EEMM1=EEM                                                        01920000
 330   CONTINUE                                                         01930000
 340   CONTINUE                                                         01940000
C                                                                       01950000
       FNMXM1=FLOAT(NMAX-1)                                             01960000
C                                                                       01961000
C        SCALE AZIMUTHAL CONTRIBUTION                                   01962000
C                                                                       01963000
       TPCWI=TPCWX/FNMXM1*DP+TPOWI                                      01970000
       POWI=POWX/FNMXM1*DP+PCWI                                         01980000
C                                                                       01981000
C        PRINT-OUT AVERAGE THETA-EDGE AND POWER THRU GRID SEGMENT       01982000
C                                                                       01983000
      WRITE(6,2304)TEDGEA,PCWI                                          01990000
C                                                                       01991000
C        PRINT-OUT MAX THETA AND POWER THRU MAX THETA (I.E.,TPOW)       01992000
C                                                                       01993000
      WRITE(6,2307)THMAX,TPCWI                                          02000000
      RAT=POWI/TPCWI*100.0                                              02010000
      WRITE(6,2308)RAT                                                  02020000
C                                                                       02030000
C      COMBINE SURFACE AND FIELD DATA TO DETERMINE VECTOR A             02040000
C       A CONTAINS THREE COORDINATE COMPONENTS.                         02050000
C                                                                       02060000
      DO 400 M=1,MMAX                                                   02070000
      DO 400 N=1,NMAX                                                   02080000
```



```
      T1=FT(M,N)*SIT(M)*HP(M,N)-FP(M,N)*HT(M,N)                         02090000
      T2=FP(M,N)*HR(M,N)+F(M,N)*SIT(M)*HP(M,N)/PC                       02100000
      T3=-FT(M,N)*SIT(M)   *HR(M,N)-F(M,N)*SIT(M)*HT(M,N)/PC            02110000
C                                                                       02111000
C        CONVERT FROM SPHERICAL TO CARTESIAN COORDINATES                02112000
C                                                                       02113000
      A(M,N,1)=T1*SIT(M)*COP(N)+T2*COT(M)*COP(N)-T3*SIP(N)              02120000
      A(M,N,2)=T1*SIT(M)*SIP(N)+T2*COT(M)*SIP(N)+T3*COP(N)              02130000
      A(M,N,3)=T1*COT(M)-T2*SIT(M)                                      02140000
 400  CONTINUE                                                          02150000
C     WRITE(6,2007)I                                                    02160000
C                                                                       02170000
C       BEGIN OUTPUT GRID LOOP. FOR EACH POINT ON OUTPUT GRID DETERMINE 02180000
C         SCATTERED FIELD CONTRIBUTION FROM ENTIRE INTEG GRID.          02190000
C                                                                       02200000
C     WRITE(6,2006)I                                                    02210000
      DO 500 K=1,KMAX                                                   02220000
      TC=PP(K)/0.017453293                                              02230000
C     WRITE(6,2011)TO                                                   02240000
      DO 500 J=1,JMAX                                                   02260000
C                                                                       02270000
C      ESTABLISH PATH LENGTH PARAMETER ON INTEGRATION GRID              02280000
C                                                                       02290000
      CALL PATHL(F,J,K,MMAX,NMAX,GAM)                                   02300000
C                                                                       02310000
C       PERFORM INTEGRATION                                             02320000
C                                                                       02330000
      CALL FINT(T,P,A,GAM,MMAX-1,NMAX-1,STOT,I,J,K,PC,IEDGE,TEDGE)      02340000
C                                                                       02350000
C       ASSIGN SCATTERED FIELDS AT OUTPUT POINT                         02360000
C                                                                       02370000
      ETTO=COTT(J)*(STOT(1)*COPP(K)+STOT(2)*SIPP(K))-STOT(3)            02380000
     &*SITT(J)                                                          02390000
      EPPO=STOT(2)*COPP(K)-STOT(1)*SIPP(K)                              02400000
      ETTO=-(0.0,1.0)*PC/6.2831854*ETTO                                 02410000
      EPPO=-(0.0,1.0)*PC/6.2831854*EPPO                                 02420000
      TO=TT(J)/0.017453293                                              02430000
      A1=REAL(ETTO)                                                     02440000
      A2=AIMAG(ETTO)                                                    02450000
      A3=REAL(EPPO)                                                     02460000
      A4=AIMAG(EPPO)                                                    02470000
C                                                                       02480000
C     CONVERT   TO POLAR COORDINATES                                    02490000
C                                                                       02500000
      CALL VECTCR(A1,A2,ETAMP,ETPHI)                                    02510000
      CALL VECTCR(A3,A4,EPAMP,EPPHI)                                    02520000
C     WRITE(6,2012)TO,ETAMP,ETPHI,EPAMP,EPPHI                           02530000
C                                                                       02540000
C        SUPERIMPOSE SCATTERED FIELD VALUE WITH VALUES CORRESPONDING    02550000
C        TO OTHER INTEG GRIDS.                                          02560000
C                                                                       02570000
      ETT(J,K)=ETT(J,K)+ETTO                                            02580000
      EPP(J,K)=EPP(J,K)+EPPO                                            02590000
 500  CONTINUE                                                          02600000
C                                                                       02610000
C        COMPUTE AVERAGE THETA-EDGE FOR ALL INTEGRATION GRIDS. SKIP     02611000
C        COMPUTATION IF THETA-EDGE NOT ON PRESENT GRID SEGMENT          02611500
C                                                                       02612000
```

```
      IF(IEDGE .EQ. 11)GO TO 650
       TEDGET=TEDGET+TEDGEA
       NGEDGE=NGEDGE+1
  650 CONTINUE
C
C       IF MORE INTEGRATION GRIDS REMAIN LOOP BACK
C
      IF(I-NG1)600,700,700
  600 CALL RESET(I,MMAX,NMAX)
      GO TO 10C
C
  700 I=I+1
C
C        TEDGEA=AVERAGE OF THE AVERAGE VALUE OF THETA-EDGE FOR EACH
C        INTEGRATION GRID SEGMENT CONTAINING THETA-EDGE
C
      TEDGEA=TEDGET/FLOAT(NGEDGE)
C
      WRITE(6,2020)
C
C       ESTABLISH DIRECT RADIATION ON OUTPUT GRID
C
C     WRITE(6,2009)
      DO 755 K=1,KMAX
      PC=PP(K)/C.017453293
C
C        IDENTIFY AZIMUTHAL ANGLE. SKIP FOR THETA(1) > 30 DEG
C
      TT1=TT(1)/DTR
      IF(TT1 .GE. 29.9)GO TO 705
      IF(FIPLOT .EQ. -1)WRITE(6,2011)PC
       IF(FIPLOT .EQ. 0)WRITE(6,2004)PC
  705 CONTINUE
C
C        INITIALIZE EDGE-POWER & TOTAL POWER
C
      EPOW=0.
      TPOW=0.
      TDIFF=                  .5*DTT *DTR
      DO 750 J=1,JMAX
C
C        EVALUATE FAR-FIELD OF SWE
C
      CALL FIELD2(J,K,ETTO,EPPO)
      TC=TT(J)/0.017453293
       SIGNTT=C.0
C
C        EVALUATE SIGN(COS(THETA))
C
       IF(COTT(J) .EQ. 0.00000)GO TO 710
        SIGNTT=COTT(J)/ABS(COTT(J))
  710  CONTINUE
C
C        CALCULATE Y-COMPONENT OF FIELD
C
      EPLANE=ETTO*SIPP(K)*SIGNTT + EPPO*COPP(K)
      A1=REAL(ETTO)
      A2=AIMAG(ETTO)
```



```
      A3=REAL(EPPO)
      A4=AIMAG(EPPO)
      A5=REAL(EPLANE)
      A6=AIMAG(EPLANE)
C
      CALL VECTOR(A1,A2,ETAMP,ETPHI)
      CALL VECTOR(A3,A4,EPAMP,EPPHI)
      CALL VECTOR(A5,A6,EYAMP,EYPHI)
      IF(FIPLOT .EQ. -1)WRITE(6,2012)TC,ETAMP,ETPHI,EPAMP,EPPHI
C
      IF((FIPLOT .EQ. -1) .OR. (FIPLOT .EQ. 0))GO TO 740
      IF(J .NE. 1)GO TO 735
C
C        LABEL PLOT WITH FIELD COMPONENT NAME
C
      IF(TT1 .GE. 29.9)GO TO 735
       IF(FIPLOT .EQ. 1)WRITE(6,2103)PC
       IF(FIPLOT .EQ. 2)WRITE(6,2104)PC
       IF(FIPLOT .EQ. 3)WRITE(6,2105)PO
  735 CONTINUE
       IF(J .GT. 41)GO TO 740
C
C        PLOT DESIRED COMPONENT
C
      IF(FIPLOT .EQ. 1)CALL EPLOT(J,EYAMP,EYPHI,41,DTT,TT1,TO)
      IF(FIPLOT .EQ. 2)CALL EPLOT(J,ETAMP,ETPHI,41,DTT,TT1,TO)
      IF(FIPLOT .EQ. 3)CALL EPLOT(J,EPAMP,EPPHI,41,DTT,TT1,TO)
  740 CONTINUE
C
C        ADD DIRECT AND SCATTERED FIELDS TO YIELD TOTAL(SCATTERED)
C        FIELDS.
C
      ETT(J,K)=ETT(J,K)+ETTO
      EPP(J,K)=EPP(J,K)+EPPO
C
C        CALCULATE TOTAL & INCIDENT POWER. IF TT1 > 30 DEG,SKIP POWER
C
       IF(TT1 .GE. 29.9)GO TO 750
       EEJ=ETAMP*ETAMP + EPAMP*EPAMP
       IF(J .EQ. 1)GO TO 745
C
C        POWER AT MIDPOINT OF ANULUS. MULTIPLY BY ANULUS WIDTH
C
       POWJ=     (EEJ+EEJM1)*SIN(TT(J)-TDIFF)
       TPOW=TPOW + POWJ*DTT
       TINTEG=TC
       IF(TINTEG .LT. TEDGEA)GO TO 743
C
C         SCALE EDGE CONTRIBUTION IF NECESSARY
C
       SCALH=(TEDGEA-TINTEG+DTT)/DTT
       IF(SCALH .LE. .01)GO TO 745
       EPOW=EPOW+SCALH*POWJ*DTT
        GO TO 745
  743  EPOW=EPOW+POWJ*DTT
  745  CONTINUE
       EEJM1=EEJ
  750 CONTINUE
```

```
C                                                                      03330000
       IF(TT1 .GE. 29.9)GO TO 755                                      03335000
      PERC=100.*EPOW/TPOW                                              03340000
      WRITE(6,2301)TEDGEA,EPOW,PERC                                    03350000
       RAT=POWI/EPOW                                                   03360000
C                                                                      03361000
C        PRINT-OUT RATIO OF NEAR TO FAR-FIELD INCIDENT POWER           03362000
C                                                                      03363000
       WRITE(6,2306)TEDGEA,RAT                                         03370000
 755  CONTINUE                                                         03380000
C                                                                      03390000
C       TRANSLATE PHASE CENTER,SCALE FIELD AMPLITUDES,AND OUTPUT       03400000
C         TOTAL FIELDS                                                 03410000
C     WRITE(6,2002)TITLE                                               03420000
      WRITE(6,2010)XT,YT,ZT,SCALE                                      03430000
C                                                                      03435000
C        PARAMETERS USED IN WRITING SCATTERED FIELDS ON DISK           03440000
C                                                                      03445000
      TTPEAK=180.0-2.*ALPHA                                            03450000
      JPEAK=INT((TTPEAK-TT1)/DTT +.1) +1                               03460000
C                                                                      03461000
C        READ-IN HALF-ANGLE FOR POWER CALCULATION                      03462000
C                                                                      03463000
      READ(5,1002)TEDGEA                                               03465000
C                                                                      03470000
      DO 795 K=1,KMAX                                                  03480000
      PO=PP(K)/0.017453293                                             03490000
      IF(FSPLOT .EQ. -1)WRITE(6,2011)PO                                03500000
       IF(FSPLOT .EQ. 0)WRITE(6,2004)PO                                03510000
       TPOW=0.                                                         03520000
       EPOW=0.                                                         03530000
C                                                                      03531000
C        TRANSLATE ANGLES TOWARDS 0.0 DEG AT BEAM PEAK                 03532000
C                                                                      03533000
       TDIFF=(180.-2.*ALPHA+.5*DTT)*DTR                                03540000
      DO 760 J=1,JMAX                                                  03550000
      TO=TT(J)/0.017453293                                             03560000
       SIGNTT=0.0                                                      03570000
       IF(COTT(J) .EQ. 0.00000)GO TO 765                               03580000
        SIGNTT=COTT(J)/ABS(COTT(J))                                    03590000
 765   CONTINUE                                                        03600000
C                                                                      03601000
C        EVALUATE Y COMPONENT OF SCATTERED FIELD                       03602000
C                                                                      03603000
      EPLANE=ETT(J,K)*SIPP(K)*SIGNTT + EPP(J,K)*COPP(K)                03610000
      A1=REAL(ETT(J,K))                                                03620000
      A2=AIMAG(ETT(J,K))                                               03630000
      A3=REAL(EPP(J,K))                                                03640000
      A4=AIMAG(EPP(J,K))                                               03650000
      A5=REAL(EPLANE)                                                  03660000
      A6=AIMAG(EPLANE)                                                 03670000
C                                                                      03680000
      CALL VECTOR(A1,A2,ETAMP,ETPHI)                                   03690000
      CALL VECTOR(A3,A4,EPAMP,EPPHI)                                   03700000
      CALL VECTOR(A5,A6,EYAMP,EYPHI)                                   03710000
C                                                                      03711000
C        TRANSLATE PHASE CENTER OF SCATTERED PATTERN                   03712000
C                                                                      03713000
```



```
      DP=XT*SITT(J)*COPP(K)+YT*SITT(J)*SIPP(K)+ZT*(COTT(J)+1.0)        03720000
      DP=DP*PC*57.29578                                                03730000
      ETPHI=ETPHI-DP                                                   03740000
      EPPHI=EPPHI-DP                                                   03750000
       EYPHI=EYPHI-DP                                                  03760000
C                                                                      03770000
C        ADJUST PHASES TO -180,180 RANGE                               03780000
C                                                                      03790000
      CALL ADJUST(ETPHI)                                               03800000
      CALL ADJUST(EPPHI)                                               03810000
      CALL ADJUST(EYPHI)                                               03820000
C                                                                      03821000
C        SCALE FIELD AMPLITUDES                                        03822000
C                                                                      03823000
      ETAMP=ETAMP*SCALE                                                03830000
      EPAMP=EPAMP*SCALE                                                03840000
      EYAMP=EYAMP*SCALE                                                03850000
C                                                                      03860000
      IF(FSPLOT .EQ. -1)GO TO 780                                      03870000
      IF(J .NE. 1)GO TO 770                                            03880000
       IF(FSPLOT .EQ. 1)WRITE(6,2106)PO                                03890000
       IF(FSPLOT .EQ. 2)WRITE(6,2107)PO                                03900000
       IF(FSPLOT .EQ. 3)WRITE(6,2108)PO                                03910000
 770  CONTINUE                                                         03920000
       IF(FSPLOT .EQ. 1)CALL EPLOT(J,EYAMP,EYPHI,JMAX,DTT,TT1,TO)      03930000
       IF(FSPLOT .EQ. 2)CALL EPLOT(J,ETAMP,ETPHI,JMAX,DTT,TT1,TO)      03940000
       IF(FSPLOT .EQ. 3)CALL EPLOT(J,EPAMP,EPPHI,JMAX,DTT,TT1,TO)      03950000
 780  CONTINUE                                                         03960000
C                                                                      03970000
C        STORE 31 SCATTERED PATTERN VALUES IN THETA,AMP,& PHASE ARRAYS. 03980000
C        CHOOSE EITHER LEFT OR RIGHT SIDE OF BEAM. TRANSLATE ANGLES    03981000
C                                                                      03990000
      IF(NDISK .EQ. 0)GO TO 790                                        03995000
      IF(NDISK .EQ. 2)GO TO 791                                        04000000
      IF((J .LT. (JPEAK-30)) .OR. (J .GT. JPEAK))GO TO 790             04010000
        L=JPEAK-J+1                                                    04020000
        TTOUT(L)=TTPEAK-TO                                             04021000
        TAMP(L)=EYAMP                                                  04021100
        TPHI(L)=EYPHI                                                  04021200
        GO TO 790                                                      04021300
 791   CONTINUE                                                        04021400
       IF((J .LT. JPEAK) .OR. (J .GT. (JPEAK+30)))GO TO 790            04021500
        L=J-JPEAK+1                                                    04021600
        TTOUT(L)=TO-TTPEAK                                             04021700
        TAMP(L)=EYAMP                                                  04021800
        TPHI(L)=EYPHI                                                  04021900
 790  CONTINUE                                                         04025000
C                                                                      04030000
C        PRINT-OUT SCATTERED PATTERN VALUES                            04031000
C                                                                      04032000
      IF(FSPLOT .EQ. -1)WRITE(6,2012)TO,ETAMP                          04040000
     &,ETPHI,EPAMP,EPPHI                                               04050000
C                                                                      04060000
C        CALCULATE POWER IN SCATTERED PATTERN                          04070000
C                                                                      04080000
       EEJ=ETAMP*ETAMP + EPAMP*EPAMP                                   04090000
       IF(J .EQ. 1)GO TO 785                                           04100000
C                                                                      04101000
```

```
C         POWER AT CENTER OF ANULUS. ADD TO TOTAL POWER
C
        PCWJ=.5*ABS((EEJ+EEJM1)*SIN(TT(J)-TDIFF))
        TPOW=TPCW+PCWJ*DTT
C
C         TRANSLATE THETA TC 0 DEG AT BEAM PEAK;< 0 DEG BEFORE PEAK;
C         >0 DEG AFTER PEAK.
C
        TINTEG=     TT(J)/CTR+2.*ALPHA-180.0
C
C         USE 2 SLIGHTLY DIFFERENT POWER ALGORITHMS,DEPENDING ON WHICH
C         SIDE OF BEAM PEAK THETA IS PRESENTLY ON.
C
       IF(TINTEG .GT. 0.01)GO TO 810
        IF(TINTEG .GT. (DTT-TEDGEA))GO TO 820
         SCALH=(TINTEG+TEDGEA)/DTT
         IF(SCALH .LE. .01)GC TO 830
         EPOW=EPOW+SCALH*POWJ*DTT
          GC TO 830
820      EPOW=EPOW+POWJ*DTT
830      CONTINUE
         EEJM1=EEJ
        GO TC 760
810     CONTINUE
        IF(TINTEG .LT. TEDGEA)GO TC 775
         SCALH=(TEDGEA-TINTEG+DTT)/DTT
         IF(SCALH .LE. .01)GO TO 785
         EPOW=EPOW+SCALH*PCWJ*DTT
          GO TC 785
775      EPOW=EPOW+POWJ*DTT
785    CONTINUE
       EEJM1=EEJ
760   CONTINUE
C
C        PRINT-CUT POWER RESULTS
C
      PERC=100.*EPOW/TPOW
      WRITE(6,2305)TEDGEA,EPOW,PERC
      RAT=EPOW/POWI
      WRITE(6,2303)TEDGEA,RAT
C
795     CONTINUE
C
C        WRITE PATTERN INTO DISK DATA SET (#12)
C
      IF(NDISK .EC. 0)STOP
      DC 840 L=1,31
840    WRITE(12,2023)TTCLT(L),TAMP(L),TPHI(L)
C
       STOP
1001  FCRMAT(18A4)
1002  FCRMAT(1CF10.4)
1003   FORMAT(5I5)
1005  FORMAT(I5,F7.2,2I5)
1004  FORMAT(7(1X,F9.5))
2001  FORMAT(1H1,'      NRAC    SCATTERING PROGRAM',/,5X,18A4)
2002  FCRMAT(1HO,5X,18A4)
2003  FCRMAT('C','     PROPAGATION CONSTANT=',F10.5)
2004  FCRMAT('O',10X,'PHI= ',F6.2,' DEGREES')
2006  FORMAT(1HO,' SCATTERED FIELDS FROM GRID',I2)
2007  FORMAT(1HO,' BEGIN INTEGRATICN OVER GRID',I2)
2009  FORMAT(1H1,' DIRECT RADIATION FROM INCIDENT FIELDS')
2010  FCRMAT(1HO'SUPERPOSITION OF ALL GRID SCATTERED FIELDS AND DIRECT',
     &' FIELDS',/,/,' PHASECENTER TRANSLATED BY X=',F8.4,'  Y= ',F8.4,
     &5H  Z= ,F8.4,/,/,
     &'  AMPLITUDE VALUES SCALED BY FACTOR OF ',F4.2)
2011  FORMAT(7HO  PHI=,F7.2,/
     &17X,7HE THETA,15X,5HE PHI,/,
     &9H    THETA,2(20H      VOLTS       PHASE))
2012  FORMAT(F9.2,F11.6,F8.2,F12.6,F8.2)
2013  FORMAT(F10.2,F10.6,F10.2,F10.6,F10.2)
2018  FORMAT('O','N    RHO-EDGE')
2019  FCRMAT(5(1H ,I5,1X,F8.4))
2020  FORMAT('O',20X,'***** END OF INTEGRATION *****',/)
2021  FORMAT('O','***** INTEGRATION GRID # ',I2,' *****      IEDGE=',I3)
2022   FORMAT('O','N   THETA EDGE (DEG.)')
2023   FORMAT(3F10.5)
2301  FORMAT('O','THE FAR-FIELD INCIDENT PCWER THRU ',F6.2,' DEGREES',
     &' IS ',F6.4,' VOLTS SQUARED = ',F6.2,' % OF TCTAL POWER')
2303  FORMAT('O','THE RATIC OF FAR-FIELD SCATTERED POWER TO NEAR-FIELD'
     &,' INCIDENT PCWER THRU ',F6.2,' DEGREES IS ',F6.4)
2304   FORMAT('O','THE NEAR-FIELD INCIDENT POWER THRU ',F6.2,
     &' DEGREES IS ',F6.4,' SQUARE VOLTS')
2305  FORMAT('O','THE FAR-FIELD SCATTERED POWER THRU ',F6.2,' DEGREES',
     &' IS ',F6.4,' VOLTS SQUARED = ',F6.2,' % OF TOTAL POWER')
2306  FORMAT('O','THE RATIO OF NEAR-FIELD TO FAR-FIELD INCIDENT POWER',
     &' THRU ',F6.2,' DEGREES IS ',F6.4)
2307  FORMAT('C','THE NEAR-FIELD INCIDENT POWER THRU ',F6.2,' DEGREES',
     &' (I.E.,THE TOTAL POWER) IS ',F6.4,' SQUARE VOLTS')
2308  FORMAT('O','THE POWER STRIKING THE REFLECTOR IS ',F6.2,' % OF',
     &' THE TOTAL POWER')
2100  FORMAT('1',26X,'RADIAL COMPONENT OF (NEAR) INCIDENT MAGNETIC',
     &' FIELC VS. PCLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2101  FORMAT('1',26X,' THETA COMPONENT OF (NEAR) INCIDENT MAGNETIC',
     &' FIELD VS. PCLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2102  FORMAT('1',27X,'    PHI COMPONENT CF (NEAR) INCIDENT MAGNETIC',
     &' FIELD VS. POLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2103  FORMAT('1',35X,'Y-COMP. OF (FAR) INCIDENT ELECTRIC FIELD VS. ',
     &'POLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2104  FORMAT('1',28X,'THETA COMPONENT OF (FAR) INCIDENT ELECTRIC FIELD'
     &,' VS. POLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2105  FORMAT('1',29X,'PHI CCMPONENT OF (FAR) INCIDENT ELECTRIC FIELD'
     &,' VS. PCLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2106  FORMAT('1',32X,'Y-CCMP. OF (FAR) SCATTERED ELECTRIC FIELD VS. ',
     &'POLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2107  FORMAT('1',26X,'THETA COMPONENT CF (FAR) SCATTERED ELECTRIC ',
     &'FIELD VS. POLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
2108  FORMAT('1',27X,'  PHI COMPONENT OF (FAR) SCATTERED ELECTRIC ',
     &'FIELD VS. PCLAR ANGLE',/,14X,'PHI= ',F6.1,' DEG.')
      END
```

MAIN IS FOLLOWED BY <u>EPLOT</u>.